Towards Self-Adjusting and Dependable Communication Networks

Prof. Dr. Stefan Schmid (TU Berlin)

"We cannot direct the wind,
but we can adjust the sails."

(Folklore)

Acknowledgements:

erc
European Research Council
Established by the European Commission

WWTF

Trend

Data-Centric Applications

Datacenters (“hyper-scale”)

+network

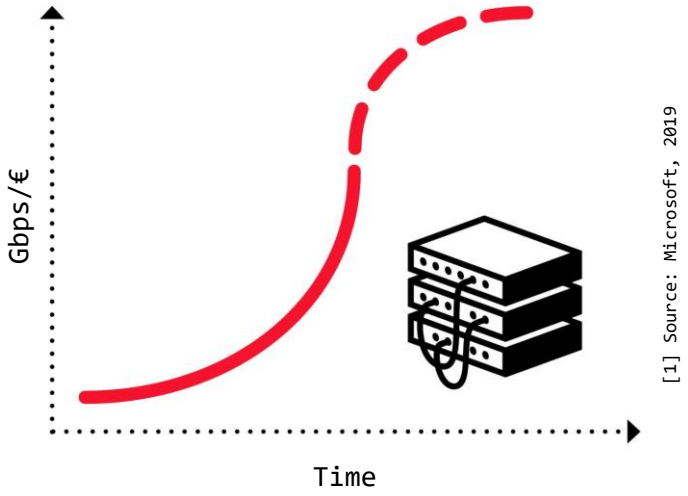
Interconnecting networks:
a **critical infrastructure**
of our digital society.

Source: Facebook

The Problem

Huge Infrastructure, Inefficient Use

- ⇢ Network equipment reaching capacity limits
  - → Transistor density rates stalling
  - → "End of **Moore's Law** in networking" [1]
- ⇢ Hence: more equipment, larger networks
- ⇢ Resource intensive and: **inefficient**

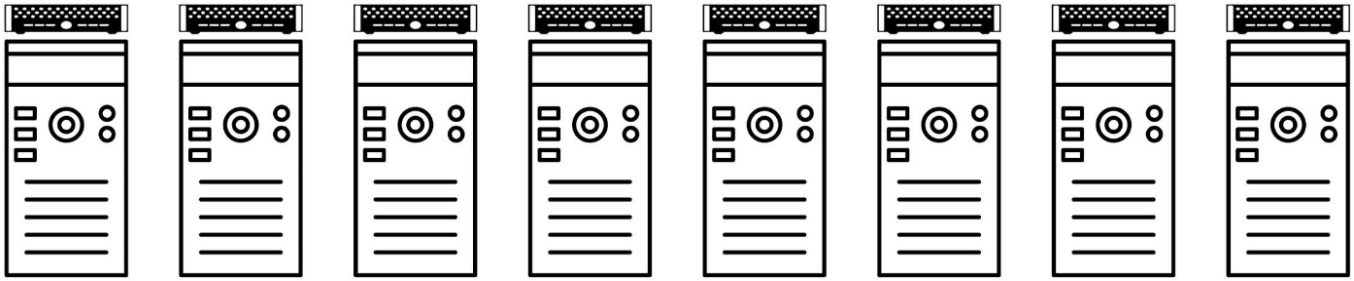


Annoying for companies,
opportunity for researchers

Root Cause

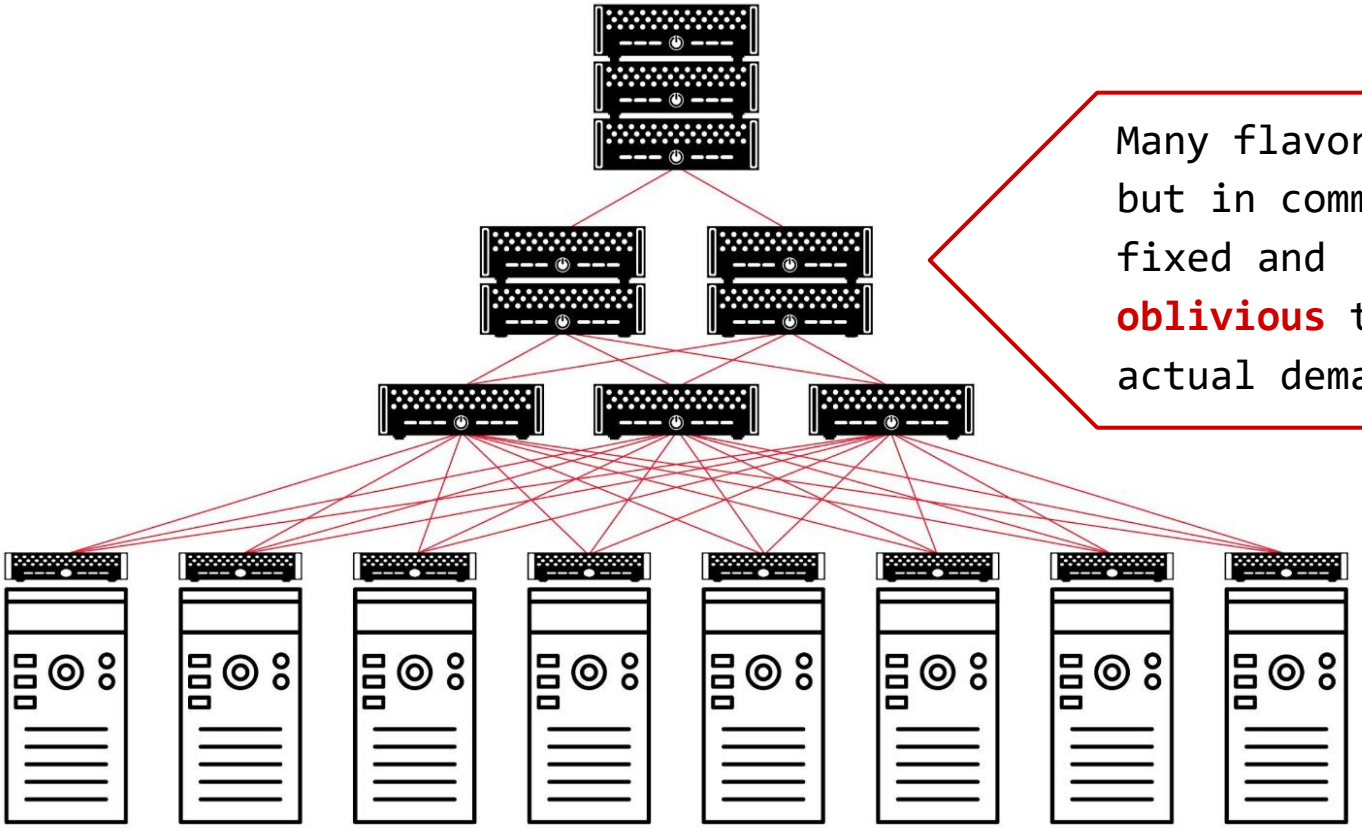
Fixed and Demand-Oblivious Topology

How to interconnect?



Root Cause

Fixed and Demand-Oblivious Topology



Many flavors, but in common: fixed and **oblivious** to actual demand.

Root Cause

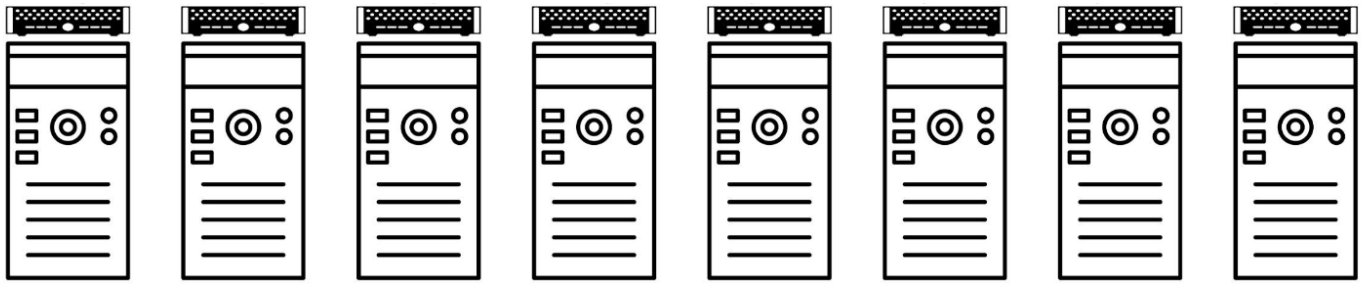
Fixed and Demand-Oblivious Topology

Highway which ignores actual traffic: frustrating!

Many flavors, but in common: fixed and **oblivious** to actual demand.

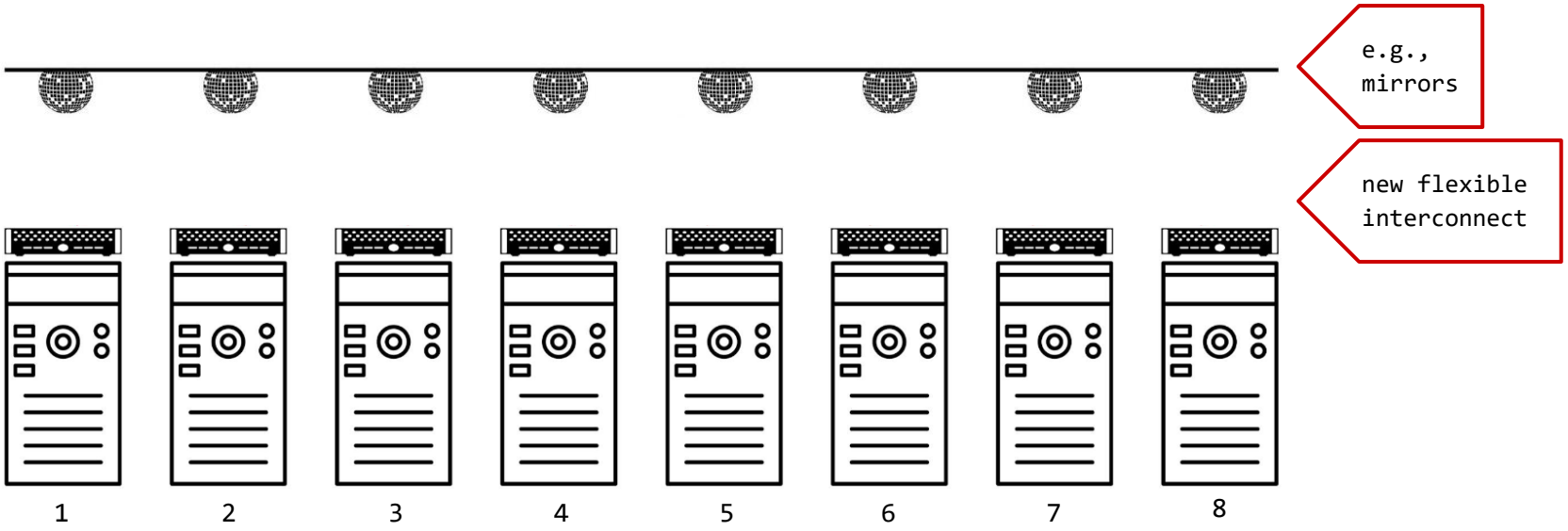
Our Vision

Flexible and Demand-Aware Topologies



Our Vision

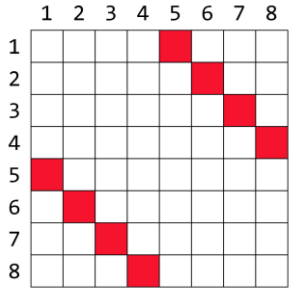
Flexible and Demand-Aware Topologies

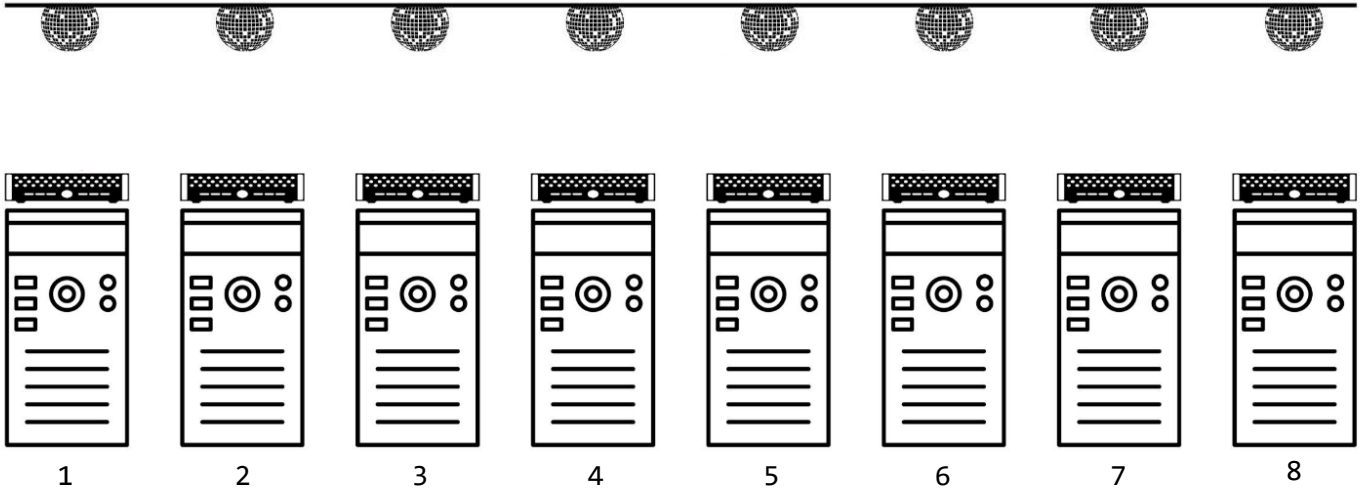


Our Vision

Flexible and Demand-Aware Topologies

demand matrix:



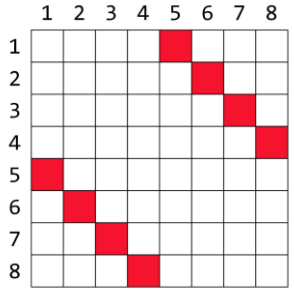


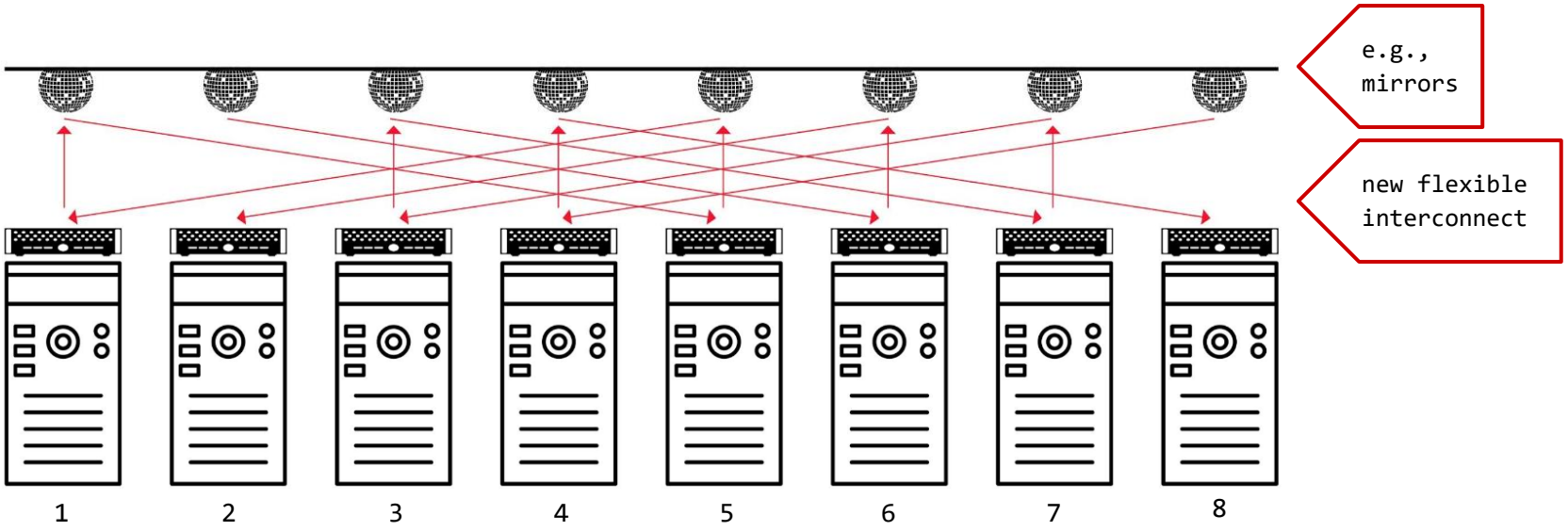
Our Vision

Flexible and Demand-Aware Topologies

Matches demand

demand matrix:

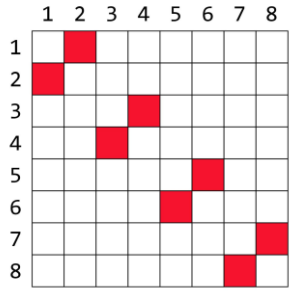


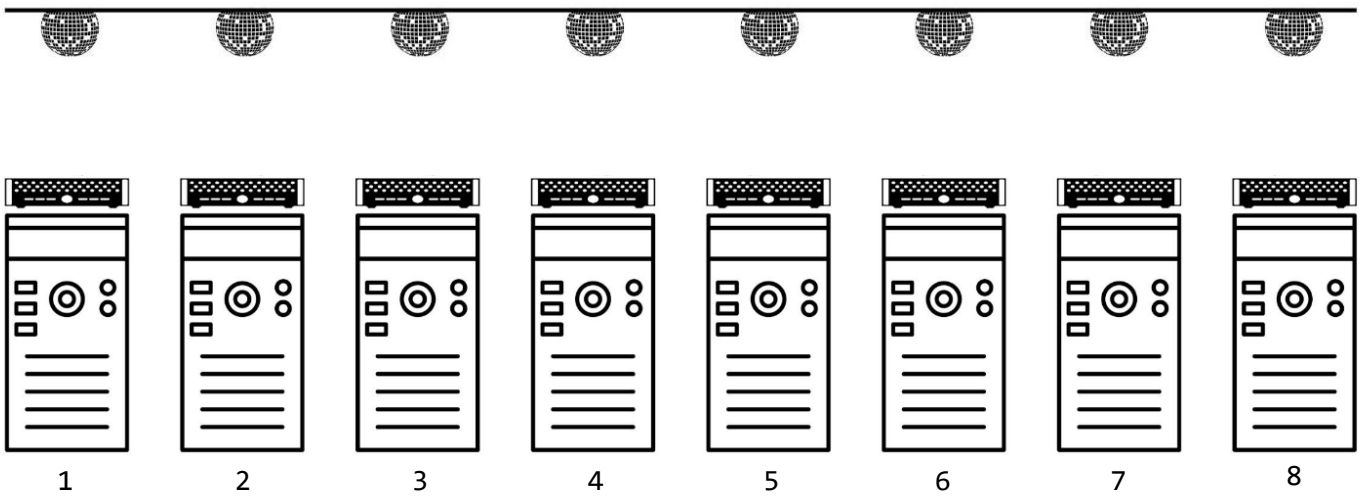


Our Vision

Flexible and Demand-Aware Topologies

new demand:



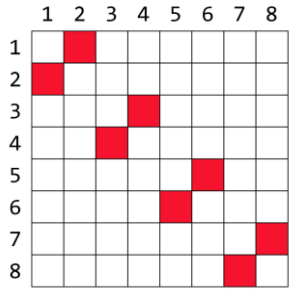


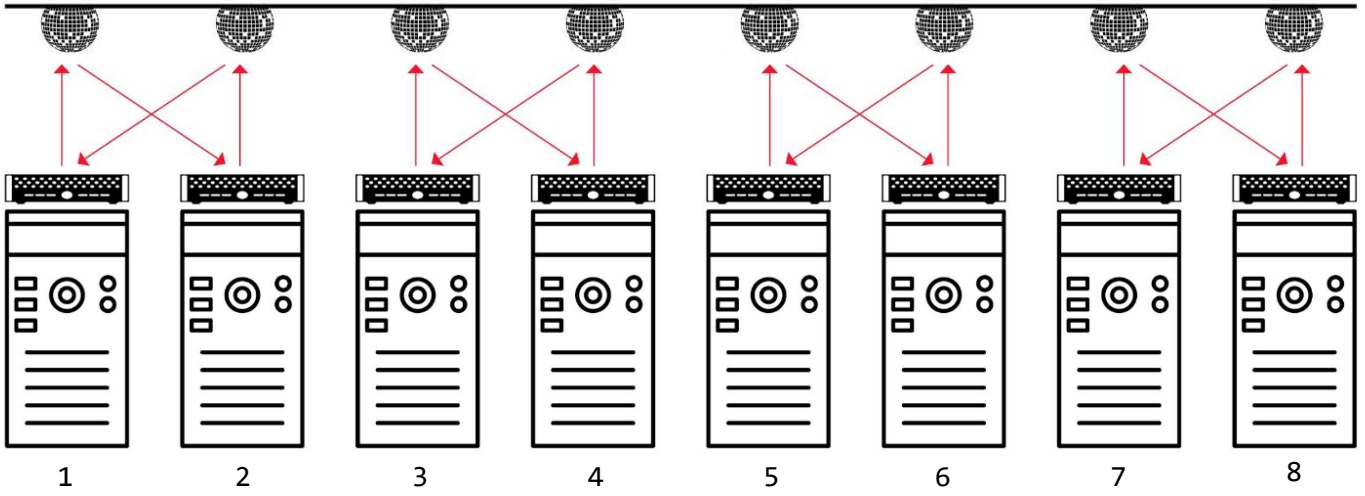
Our Vision

Flexible and Demand-Aware Topologies

Matches demand

new demand:





e.g., mirrors

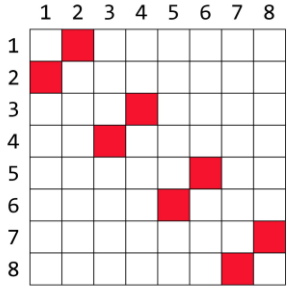
new flexible interconnect

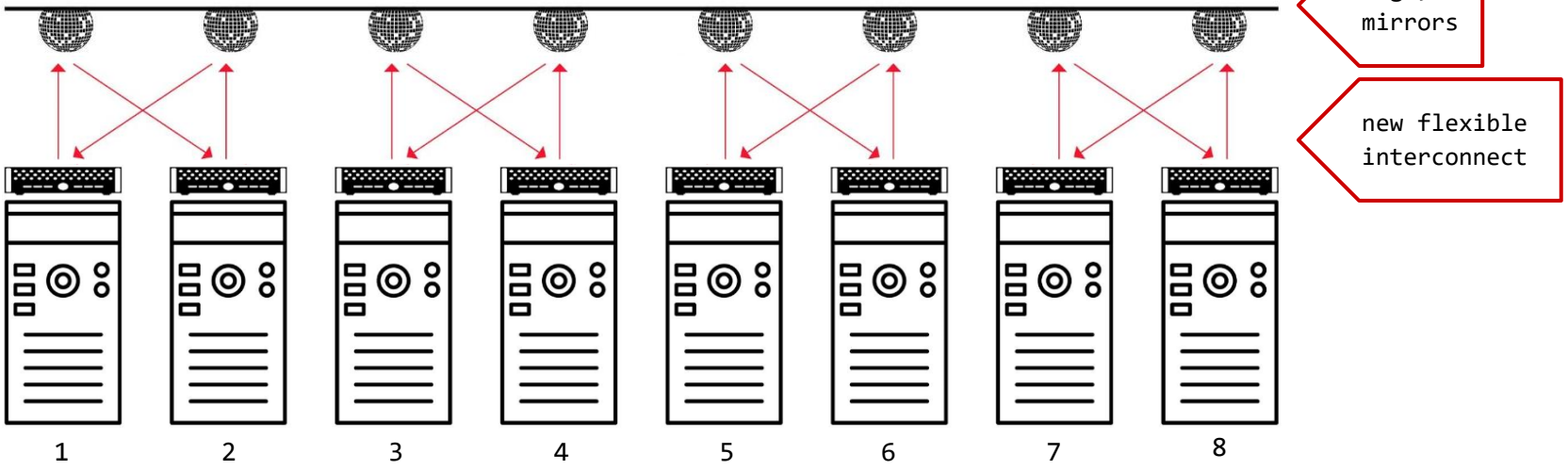
Our Vision

Flexible and Demand-Aware Topologies

Self-Adjusting Networks

new demand:



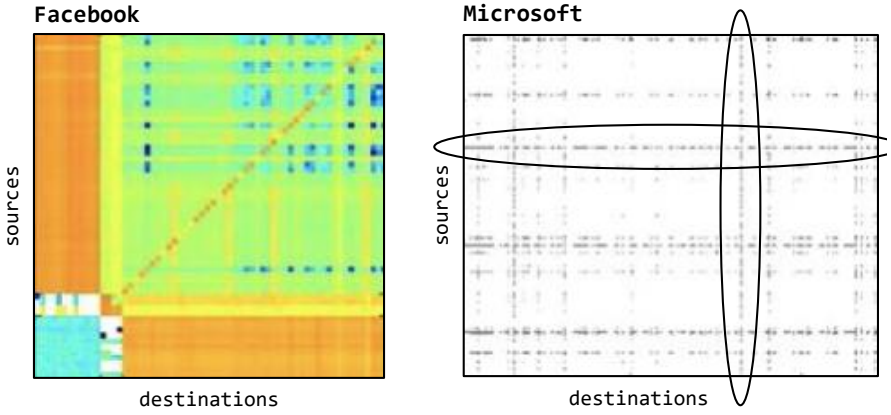


Our Motivation

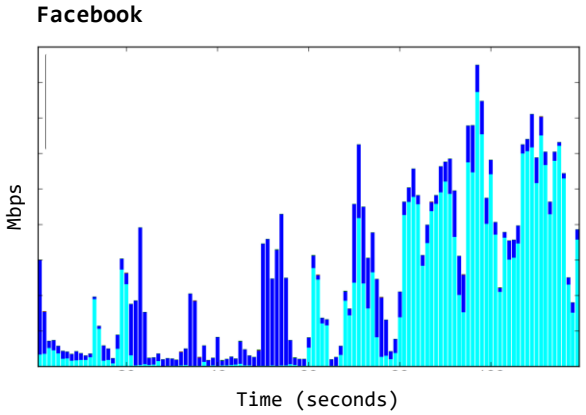
Much Structure in the Demand

Empirical studies:

traffic matrices sparse and skewed

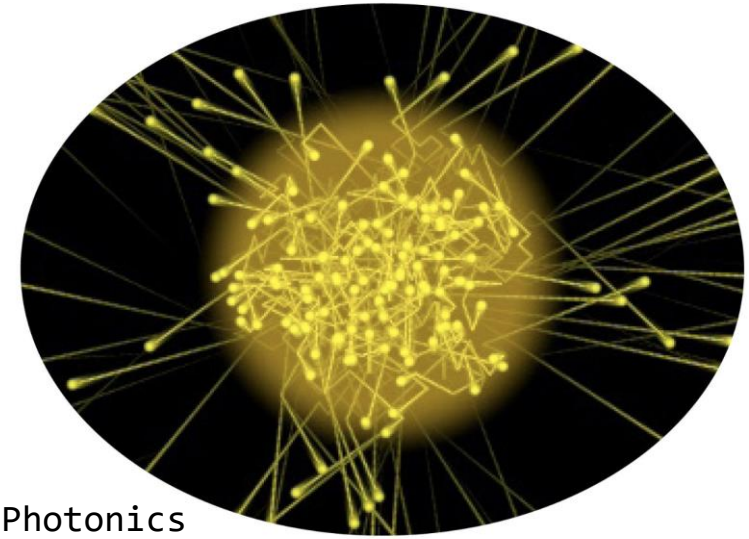


traffic bursty over time



My hypothesis: can be exploited.

Sounds Crazy? Emerging Enabling Technology.



Photonics

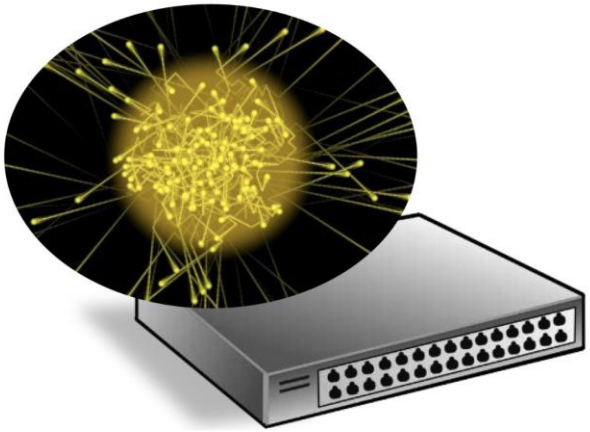
H2020:
“Photonics one of only five key enabling technologies for future prosperity.”

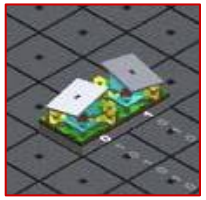
US National Research Council:
“Photons are the new Electrons.”

Enabler

Novel Reconfigurable Optical Switches

- **Spectrum** of prototypes
  - Different sizes, different reconfiguration times
  - From our last years' ACM **SIGCOMM** workshop OptSys





Prototype 1

Prototype 2

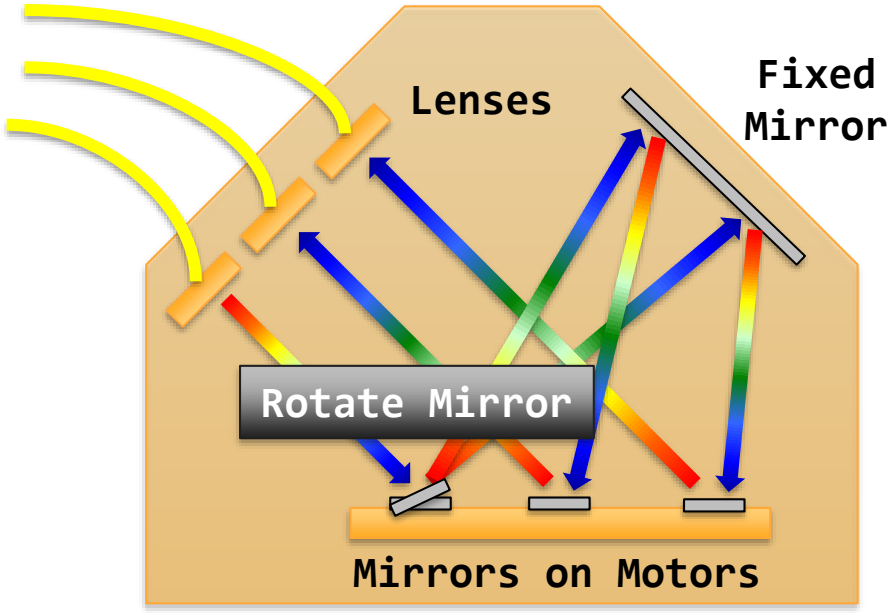
Prototype 3

Example

Optical Circuit Switch

⇢ Optical Circuit Switch rapid adaption of physical layer

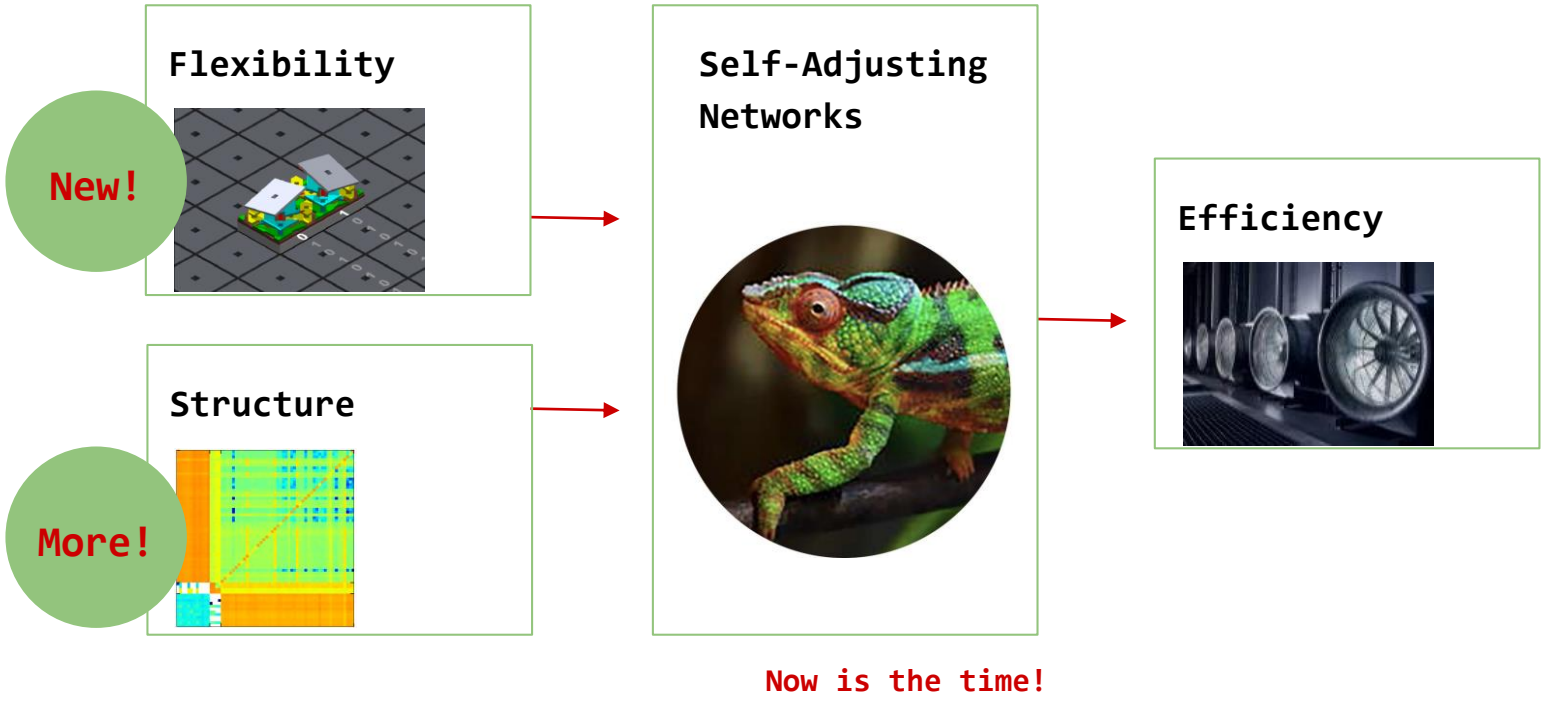
→ Based on rotating mirrors



Optical Circuit Switch

By Nathan Farrington, SIGCOMM 2010

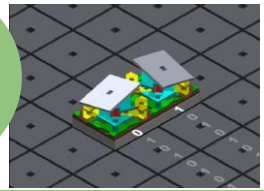
The Big Picture



The Big Picture

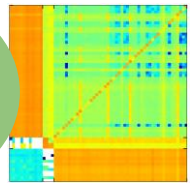
New!

Flexibility



More!

Structure



Self-Adjusting Networks

Efficiency

Now is the time!

Our goal: Develop the **foundations** of demand-aware, self-adjusting networks.

Unique Position

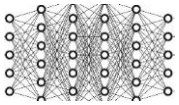
Demand-Aware, Self-Adjusting Systems

Everywhere, but mainly in software

Algorithmic trading

NETFLIX

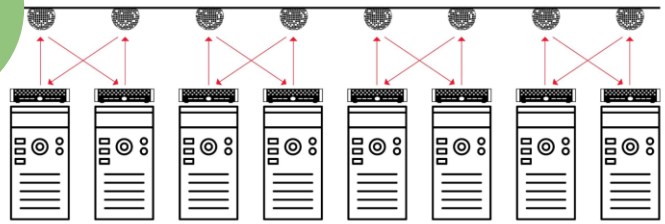
Recommender systems



Neural networks

VS

Our focus: in hardware

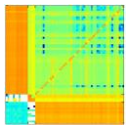


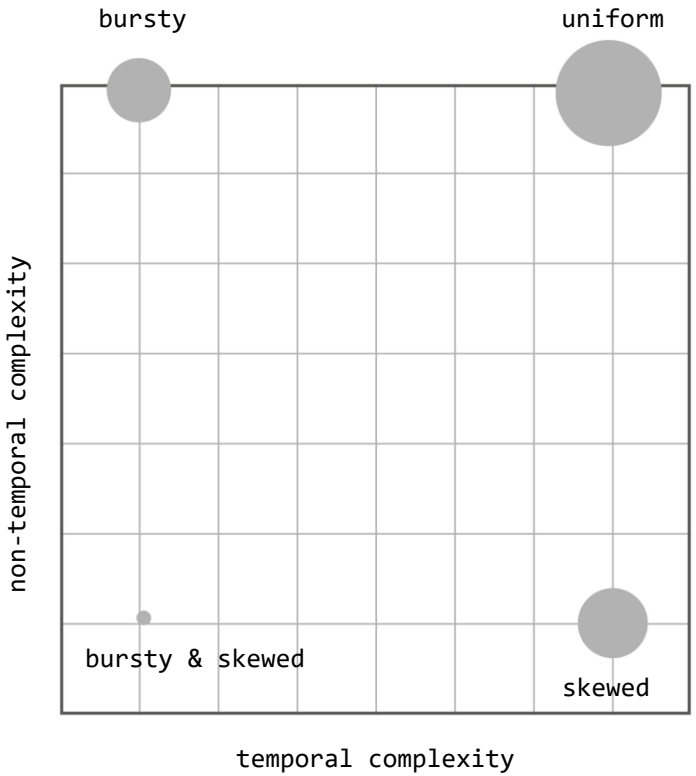
Question 1:

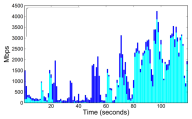
How to Quantify such “Structure” in the Demand?

An Information-Theoretic Approach

Complexity Map



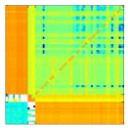


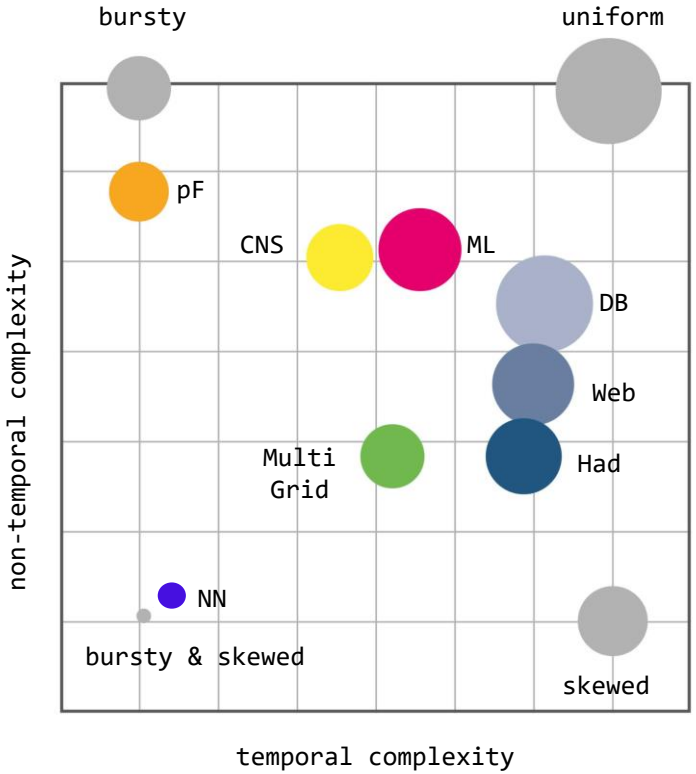


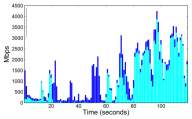
Our **approach**: iterative **randomization and compression** of trace to identify dimensions of structure.

An Information-Theoretic Approach

Complexity Map





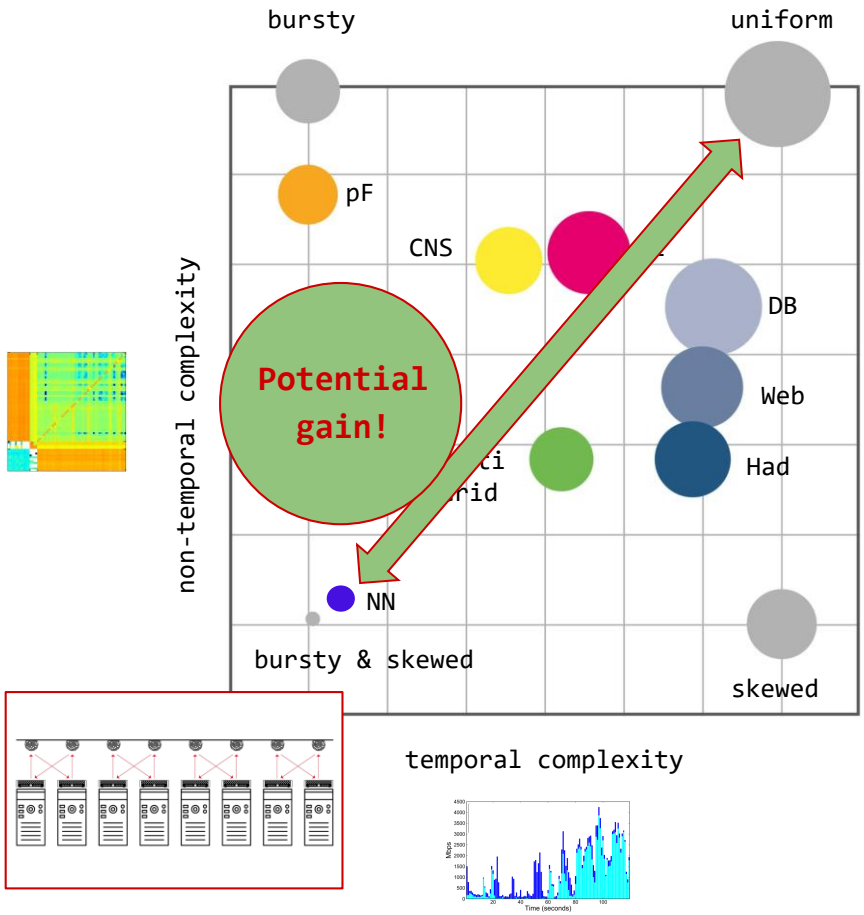


Our **approach**: iterative **randomization and compression** of trace to identify dimensions of structure.

Different structures!

An Information-Theoretic Approach

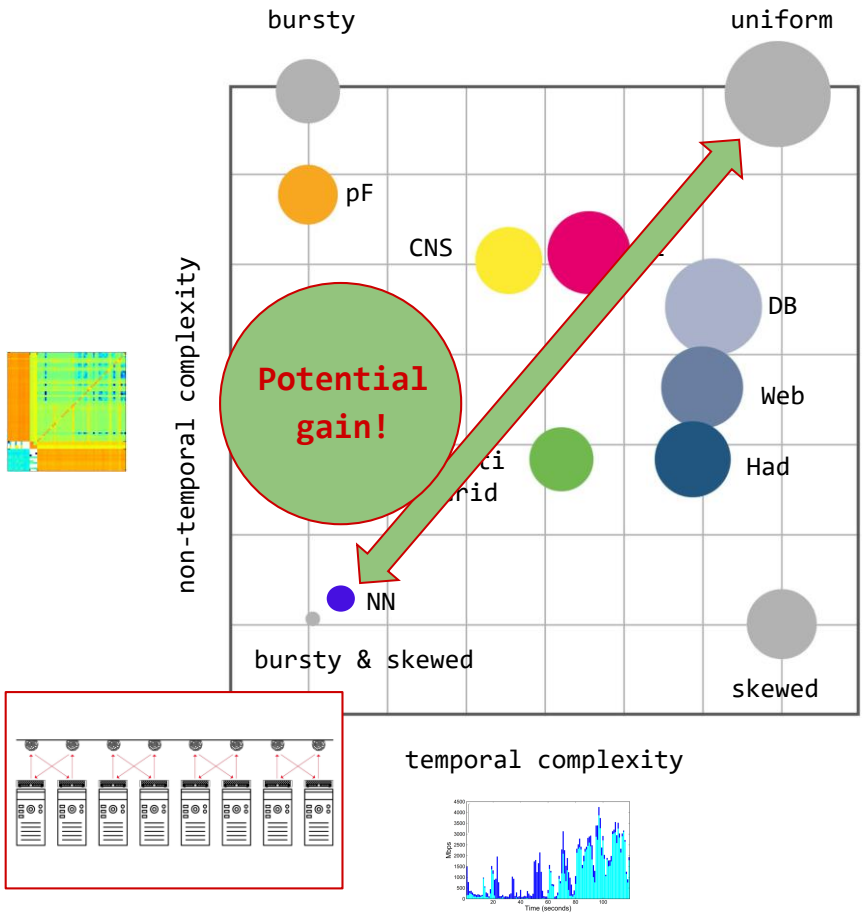
Complexity Map



Our **approach**: iterative **randomization and compression** of trace to identify dimensions of structure.

An Information-Theoretic Approach

Complexity Map



Our **approach**: iterative **randomization and compression** of trace to identify dimensions of structure.

ACM SIGMETRICS 2022

Question 2:

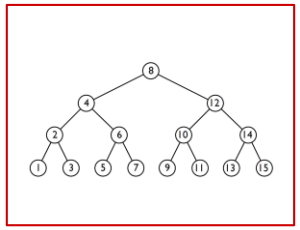
Given This Structure, What Can Be Achieved? Metrics and Algorithms?

A first insight: entropy of the demand.

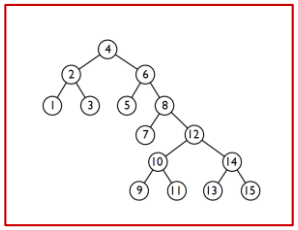
Our Approach:

Connection to Datastructures

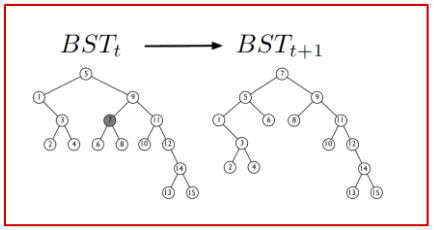
Traditional BST



Demand-aware BST



Self-adjusting BST

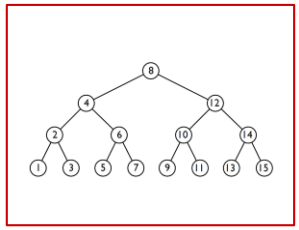


More structure: improved **access cost**

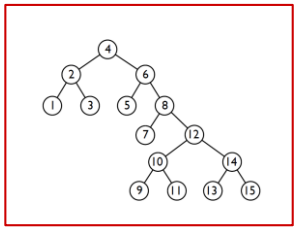
Our Approach:

Connection to Datastructures & Coding

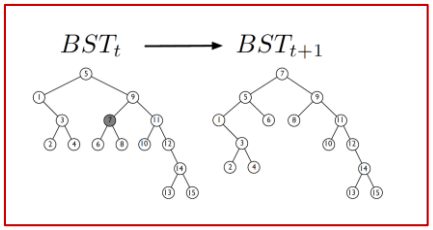
Traditional BST
(Worst-case coding)



Demand-aware BST
(Huffman coding)



Self-adjusting BST
(Dynamic Huffman coding)

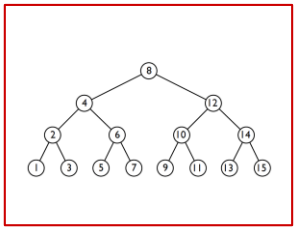


More structure: improved **access cost** / shorter **codes**

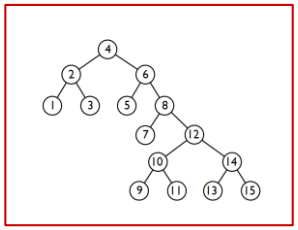
Our Approach:

Connection to Datastructures & Coding

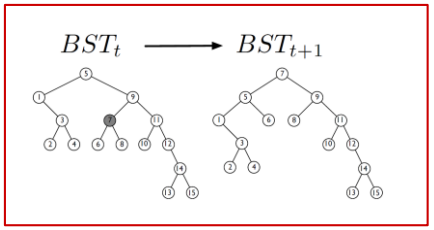
Traditional BST
(Worst-case coding)



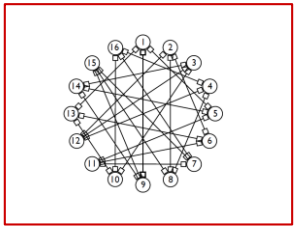
Demand-aware BST
(Huffman coding)

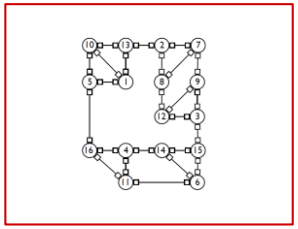


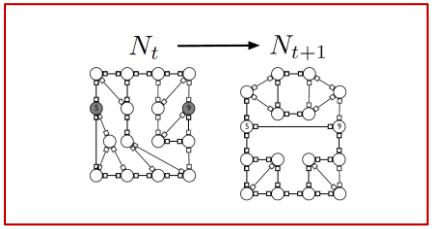
Self-adjusting BST
(Dynamic Huffman coding)



More structure: improved **access cost** / shorter **codes**





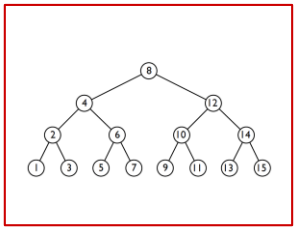


Similar **benefits**?

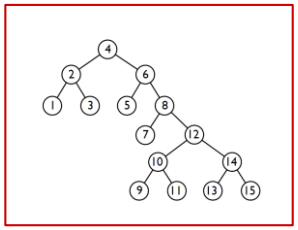
Our Approach:

Connection to Datastructures & Coding

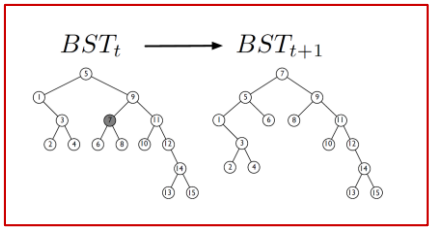
Traditional BST
(Worst-case coding)



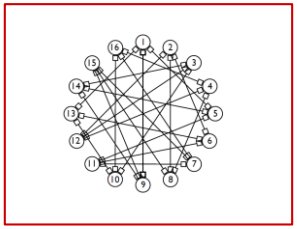
Demand-aware BST
(Huffman coding)

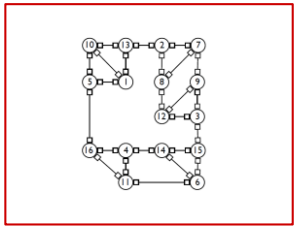


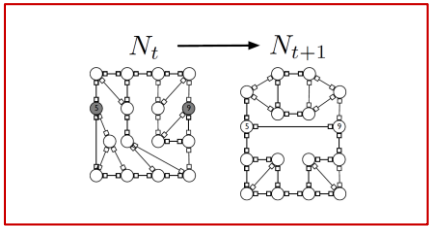
Self-adjusting BST
(Dynamic Huffman coding)



More structure: improved **access cost** / shorter **codes**







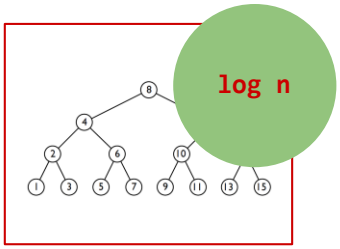
Similar **benefits**?

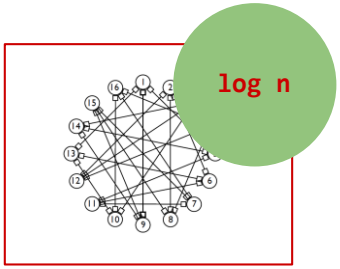
More than
an analogy!

Our Approach:

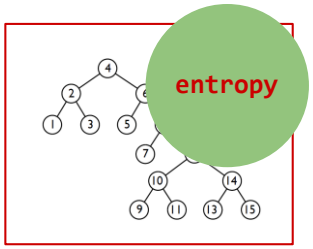
Connection to Datastructures & Coding

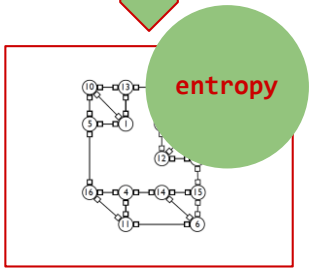
Traditional BST
(Worst-case coding)



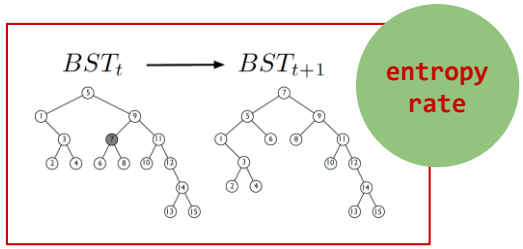


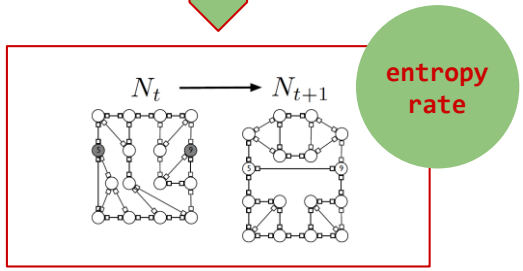
Demand-aware BST
(Huffman coding)





Self-adjusting BST
(Dynamic Huffman coding)





Reduced expected **route lengths**!

More than an analogy!

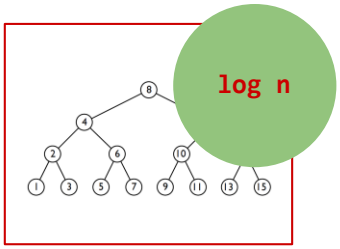
Generalize methodology:
... and transfer entropy bounds and algorithms of data-structures to networks.

Results, e.g.:
Demand-aware networks of asymptotically optimal route lengths.

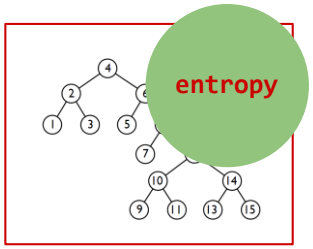
Our Approach:

Connection to Datastructures & Coding

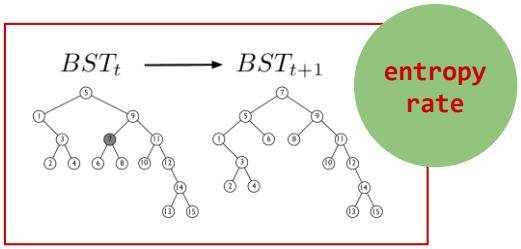
Traditional BST
(Worst-case coding)

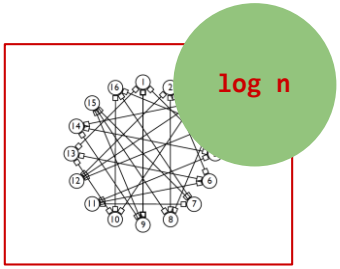


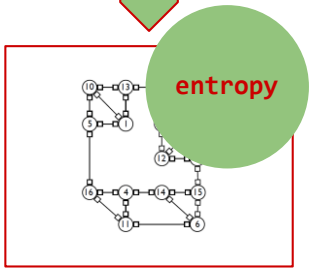
Demand-aware BST
(Huffman coding)

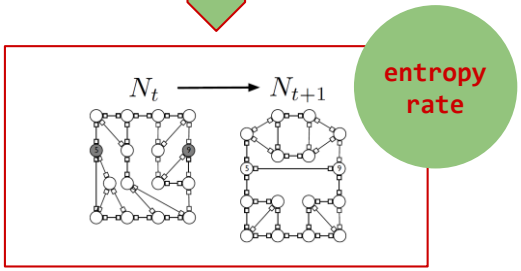


Self-adjusting BST
(Dynamic Huffman coding)









Reduced expected **route lengths**!

More than an analogy!

Can also optimize congestion and throughput:

SIGMETRICS, SIGCOMM CCR, ACM SC, NSDI, Infocom…

Dependability

Another Benefit of Self-Adjusting Networks

Requirements vs Reality

Communication networks are a critical backbone, but:

Countries disconnected

Data Centre ▸ Networks

Google routing blunder sent Japan's Internet dark on Friday

Another big BGP blunder

By Richard Chirgwin 27 Aug 2017 at 22:35 40 SHARE ▼

Last Friday, someone in Google fat-thumbed a border gateway protocol (BGP) advertisement and sent Japanese Internet traffic into a black hole.

The trouble began when The Chocolate Factory "leaked" a big route table to Verizon, the result of which was traffic from Japanese giants like NTT and KDDI was sent to Google on the expectation it would be treated as transit.

Passengers stranded

British Airways' latest Total Inability To Support Upwardness of Planes* caused by Amadeus system outage

Stuck on the ground awaiting a load sheet? Here's why

By Gareth Corfield 19 Jul 2018 at 11:16 109 SHARE ▼

Even 911 affected

Officials: Human error to blame in Minn. 911 outage

According to a press release, CenturyLink told department of public safety that human error by an employee of a third party vendor was to blame for the outage

Aug 16, 2018

Duluth News Tribune

SAINT PAUL, Minn. — The Minnesota Department of Public Safety Emergency Communication Networks division was told by its 911 provider that an Aug. 1 outage was caused by human error.

Requirements vs Reality

Communication networks are a critical backbone, but:

Countries disconnected

Data Centre ▸ Networks

Google routing blunder sent Japan's Internet dark on Friday

Another big BGP blunder

By Richard Chirgwin 27 Aug 2017 at 22:35 40 SHARE ▼

Last Friday, someone in Google fat-thumbed a border gateway protocol (BGP) advertisement and sent Japanese Internet traffic into a black hole.

The trouble began when The Chocolate Factory "leaked" a big route table to Verizon, the result of which was traffic from Japanese giants like NTT and KDDI was sent to Google on the expectation it would be treated as transit.

Passengers stranded

British Airways' latest Total Inability To Support Upwardness of Planes* caused by Amadeus system outage

Stuck on the ground awaiting a load sheet? Here's why

By Gareth Corfield 19 Jul 2018 at 11:16 109 SHARE ▼

Even 911 affected

Officials: Human error to blame in Minn. 911 outage

According to a press release, CenturyLink told department of public safety that human error by an employee of a third party vendor was to blame for the outage

Aug 16, 2018

Duluth News Tribune

SAINT PAUL, Minn. — The Minnesota Department of Public Safety Emergency Communication Networks division was told by its 911 provider that an Aug. 1 outage was caused by human error.

Even tech-savvy companies struggle:

Slide credits: Nate Foster and Laurent Vanbever

Requirements vs Reality

Communication networks are a critical backbone, but:

Countries disconnected

Data Centre ▸ Networks

Google routing blunder sent Japan's Internet dark on Friday

Another big BGP blunder

By Richard Chirgwin 27 Aug 2017 at 22:35 40 SHARE ▼

Last Friday, someone in Google fat-thumbed a border gateway protocol (BGP) advertisement and sent Japanese Internet traffic into a black hole.

The trouble began when The Chocolate Factory "leaked" a big route table to Verizon, the result of which was traffic from Japanese giants like NTT and KDDI was sent to Google on the expectation it would be treated as transit.

Passengers stranded

British Airways' latest Total Inability To Support Upwardness of Planes* caused by Amadeus system outage

Stuck on the ground awaiting a load sheet? Here's why

By Gareth Corfield 19 Jul 2018 at 11:16 109 SHARE ▼

Even 911 affected

Officials: Human error to blame in Minn. 911 outage

According to a press release, CenturyLink told department of public safety that human error by an employee of a third party vendor was to blame for the outage

Aug 16, 2018

Duluth News Tribune

SAINT PAUL, Minn. — The Minnesota Department of Public Safety Emergency Communication Networks division was told by its 911 provider that an Aug. 1 outage was caused by human error.

Mainly: human errors!

Even tech-savvy companies struggle:

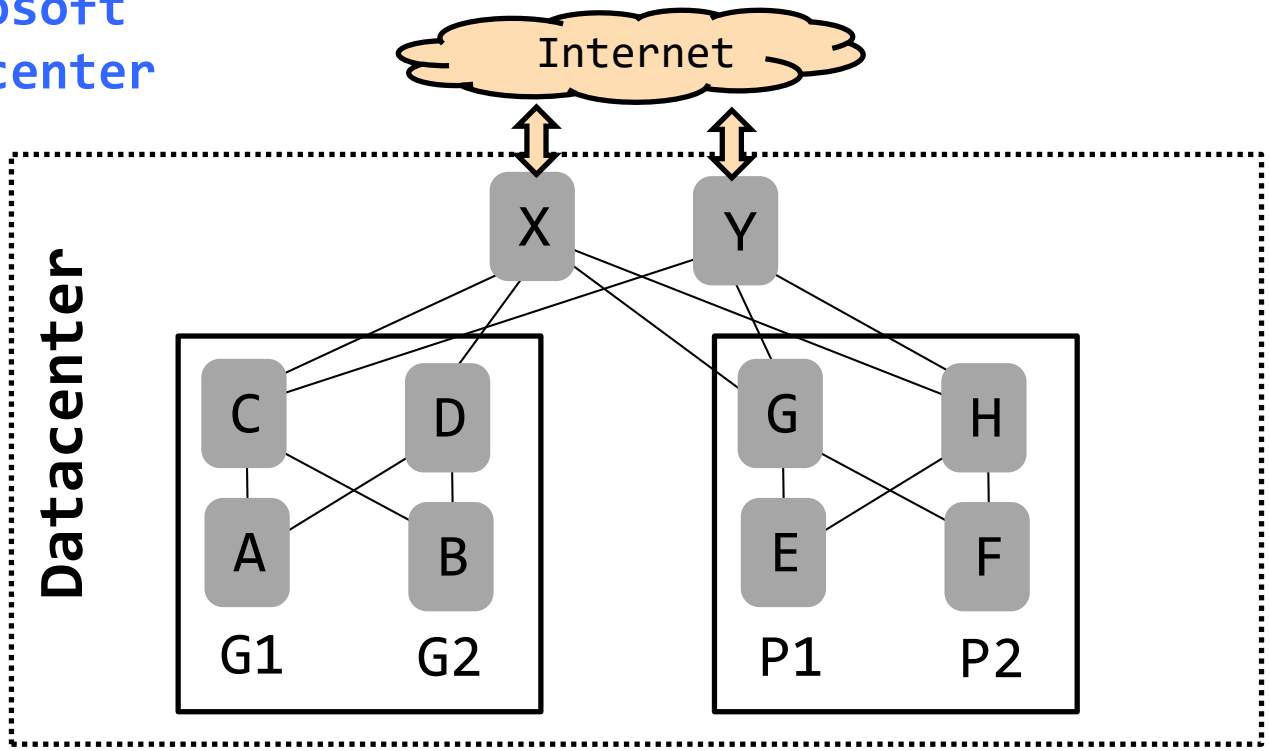
Go Daddy.com github SOCIAL CODING United Airlines amazon web services

Slide credits: Nate Foster and Laurent Vanbever

Reason: Complexity

Especially Under Failures (Policy Compliance)

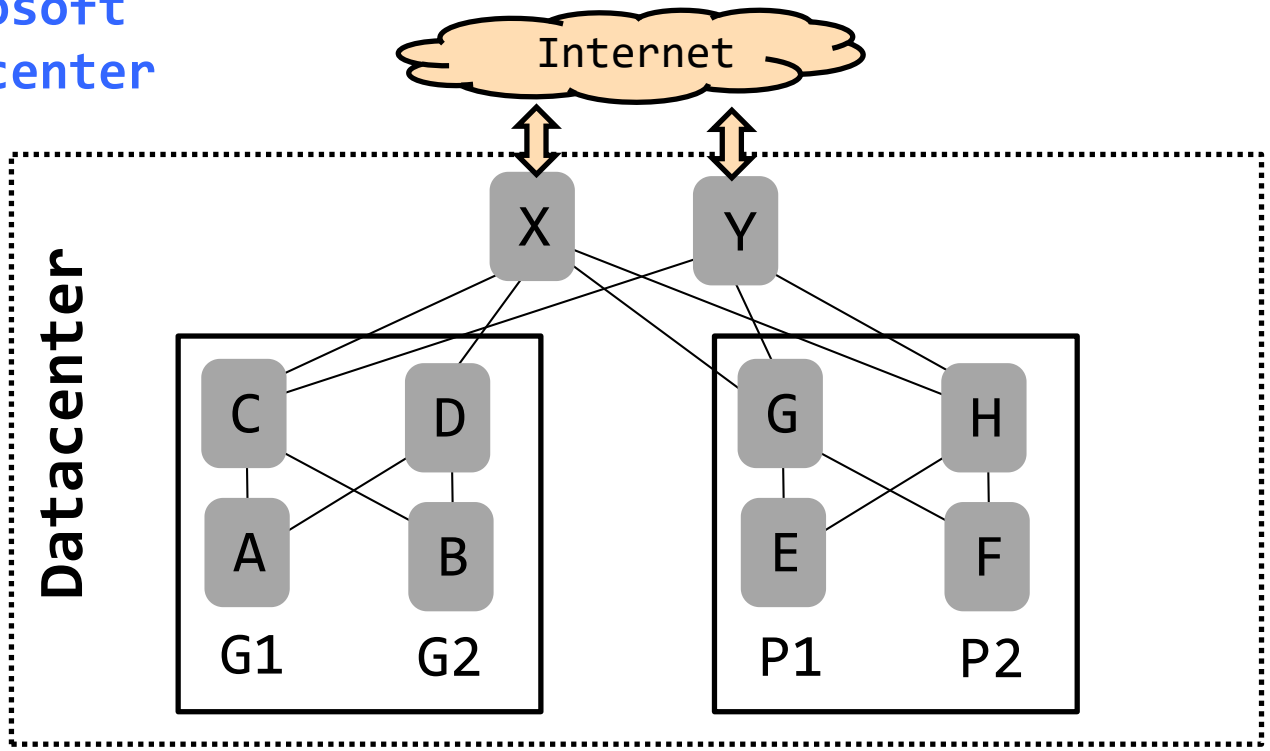
Example: BGP in **Microsoft datacenter**



Reason: Complexity

Especially Under Failures (Policy Compliance)

Example: BGP in **Microsoft datacenter**



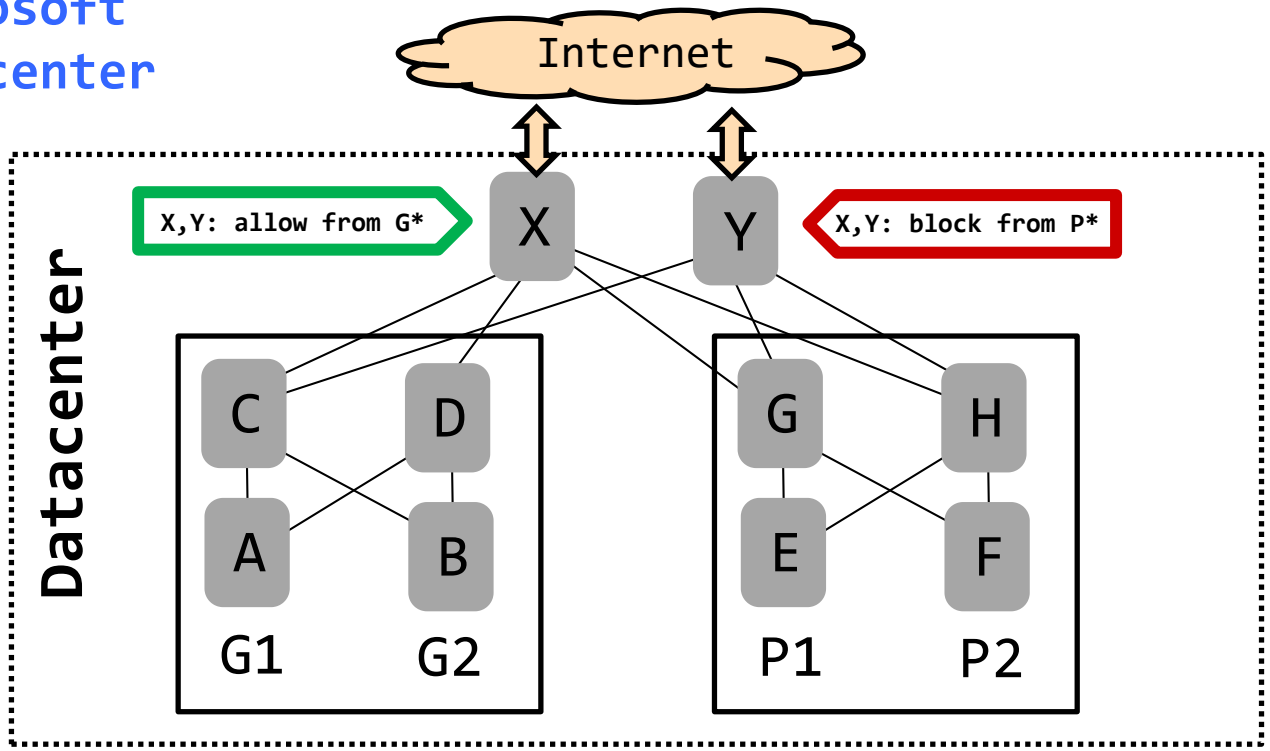
Cluster with globally reachable services

Cluster with internally accessible services

Reason: Complexity

Especially Under Failures (Policy Compliance)

Example: BGP in **Microsoft datacenter**



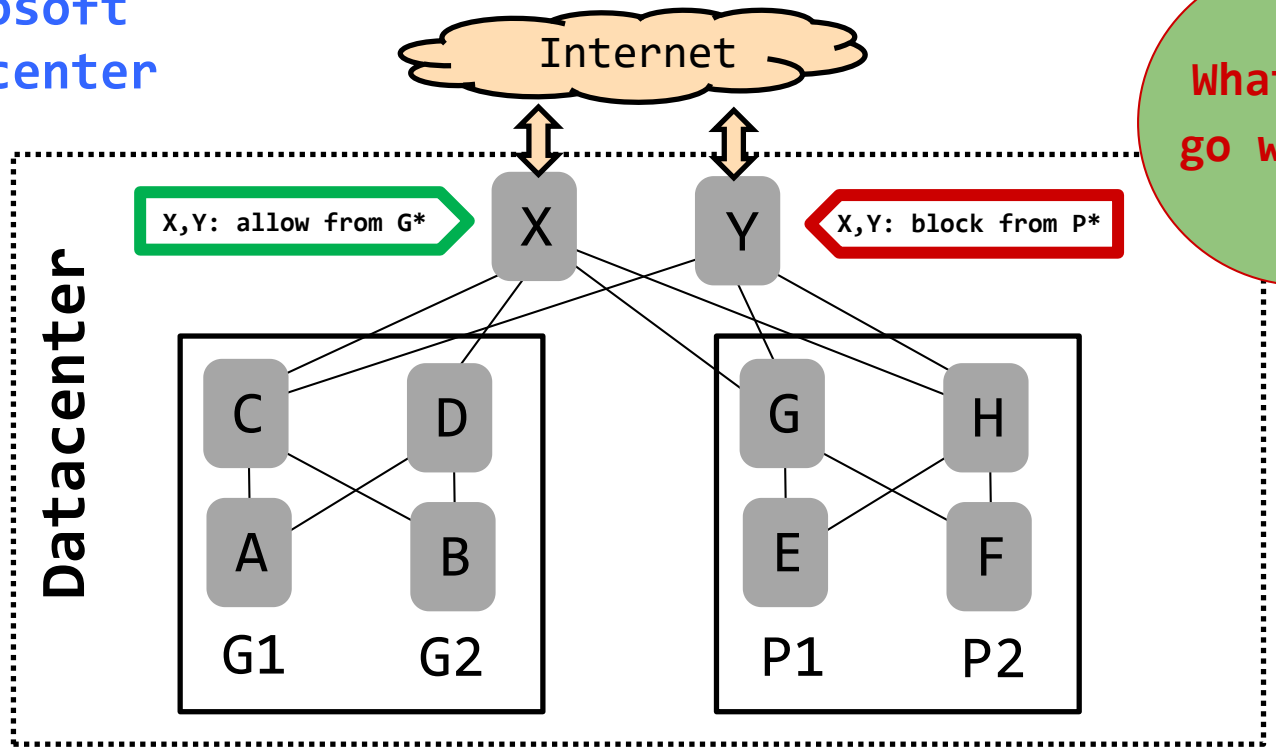
Cluster with globally reachable services

Cluster with internally accessible services

Reason: Complexity

Especially Under Failures (Policy Compliance)

Example: BGP in **Microsoft datacenter**



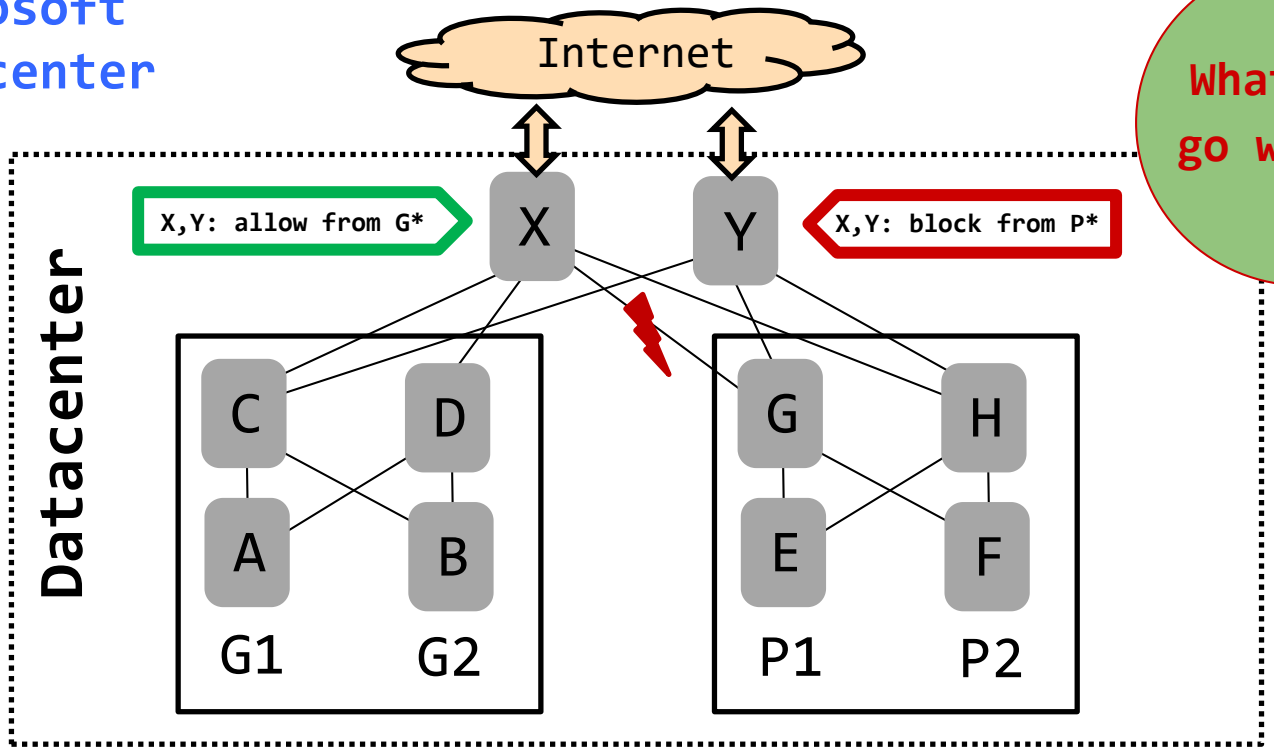
Cluster with globally reachable services

Cluster with internally accessible services

Reason: Complexity

Especially Under Failures (Policy Compliance)

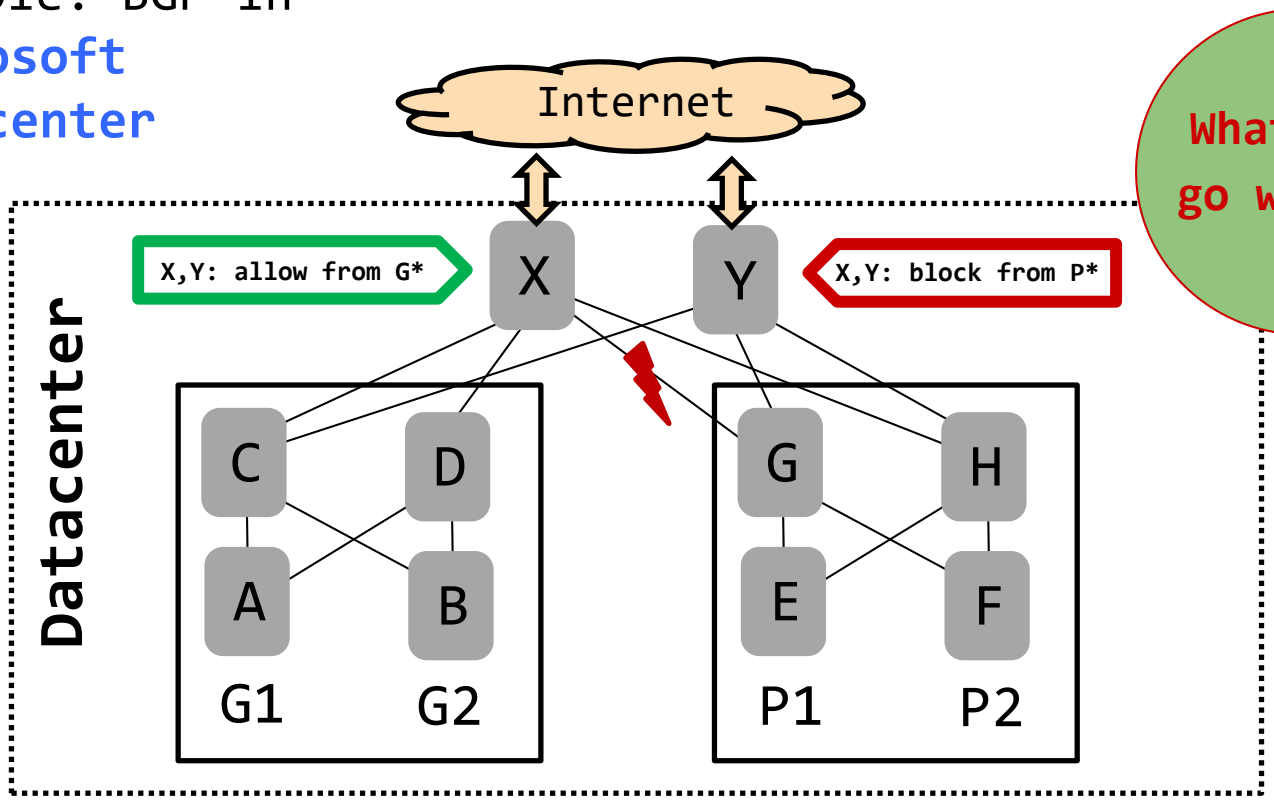
Example: BGP in **Microsoft datacenter**



Reason: Complexity

Especially Under Failures (Policy Compliance)

Example: BGP in **Microsoft datacenter**

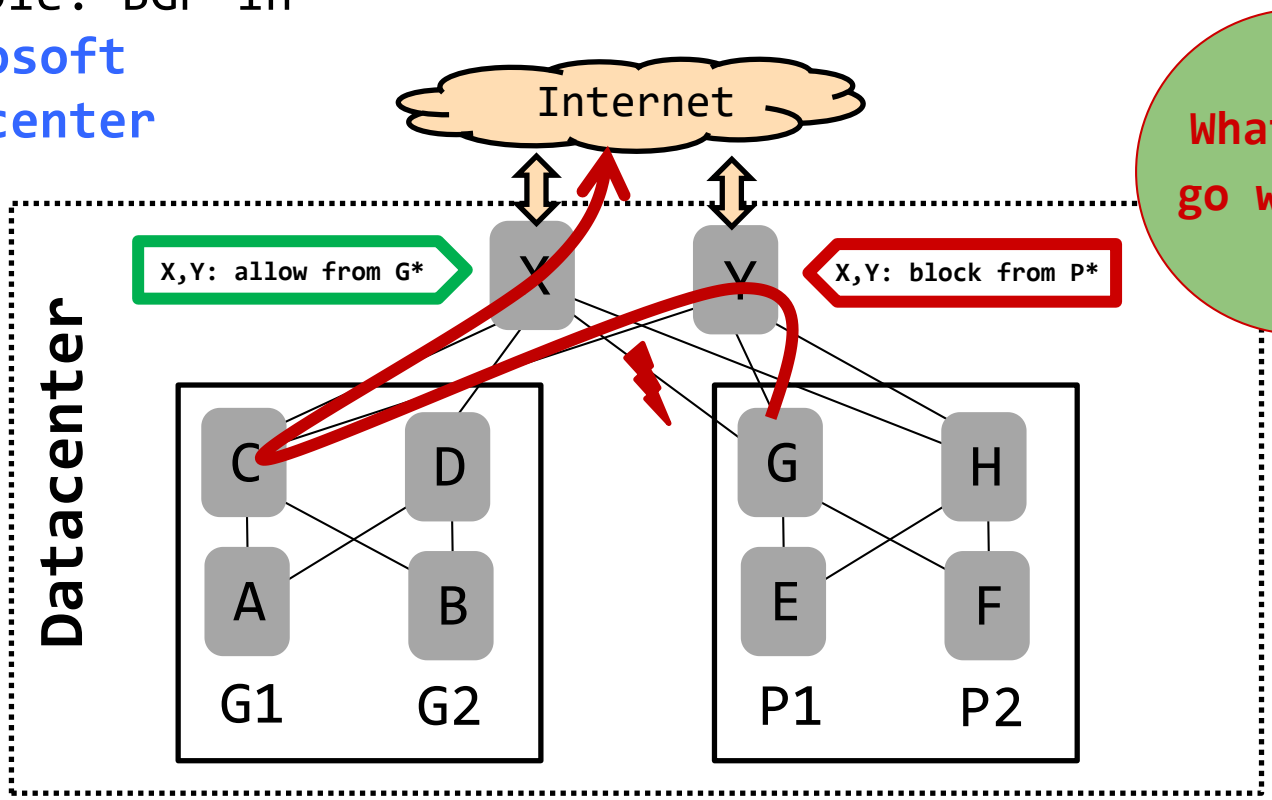


If link (G,X) fails and traffic from G is rerouted via Y and C to X:
X announces (does not block) G and H as it comes from C. (Note: BGP.)

Reason: Complexity

Especially Under Failures (Policy Compliance)

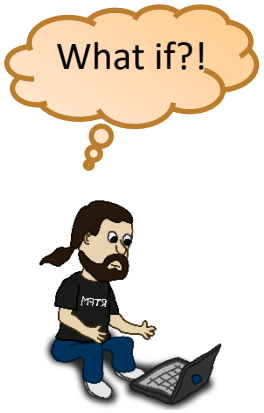
Example: BGP in **Microsoft datacenter**



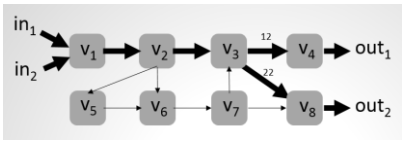
If link (G,X) fails and traffic from G is rerouted via Y and C to X:
X announces (does not block) G and H as it comes from C. (Note: BGP.)

Our Approach:

Automated Whatif Analysis



FT	In-I	In-Label	Out-I	op
τ_{v_1}	in_1	$\perp$	(v_1, v_2)	$push(10)$
	in_2	$\perp$	(v_1, v_2)	$push(20)$
τ_{v_2}	(v_1, v_2)	10	(v_2, v_3)	$swap(11)$
	(v_1, v_2)	20	(v_2, v_3)	$swap(21)$
τ_{v_3}	(v_2, v_3)	11	(v_3, v_4)	$swap(12)$
	(v_2, v_3)	21	(v_3, v_8)	$swap(22)$
	(v_7, v_3)	11	(v_3, v_4)	$swap(12)$
	(v_7, v_3)	21	(v_3, v_8)	$swap(22)$
τ_{v_4}	(v_3, v_4)	12	out_1	pop
τ_{v_5}	(v_2, v_5)	40	(v_5, v_6)	pop
τ_{v_6}	(v_2, v_6)	30	(v_6, v_7)	$swap(31)$
	(v_5, v_6)	30	(v_6, v_7)	$swap(31)$
	(v_5, v_6)	61	(v_6, v_7)	$swap(62)$
	(v_5, v_6)	71	(v_6, v_7)	$swap(72)$
τ_{v_7}	(v_6, v_7)	31	(v_7, v_3)	pop
	(v_6, v_7)	62	(v_7, v_3)	$swap(11)$
	(v_6, v_7)	72	(v_7, v_8)	$swap(22)$
τ_{v_8}	(v_3, v_8)	22	out_2	pop
	(v_7, v_8)	22	out_2	pop



local FFT	Out-I	In-Label	Out-I	op
τ_{v_2}	(v_2, v_3)	11	(v_2, v_6)	$push(30)$
	(v_2, v_3)	21	(v_2, v_6)	$push(30)$
	(v_2, v_6)	30	(v_2, v_5)	$push(40)$
global FFT	**Out-I**	**In-Label**	**Out-I**	**op**
τ'_{v_2}	(v_2, v_3)	11	(v_2, v_6)	$swap(61)$
	(v_2, v_3)	21	(v_2, v_6)	$swap(71)$
	(v_2, v_6)	61	(v_2, v_5)	$push(40)$
	(v_2, v_6)	71	(v_2, v_5)	$push(40)$

Router **configurations**
(Cisco, Juniper, etc.)

Compilation

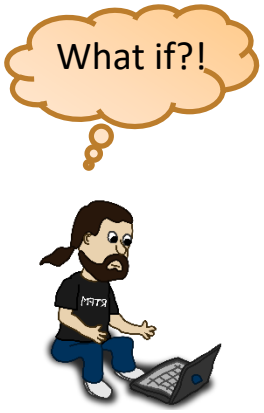
Interpretation

$pX \Rightarrow qXX$
$pX \Rightarrow qYX$
$qY \Rightarrow rYY$
$rY \Rightarrow r$
$rX \Rightarrow pX$

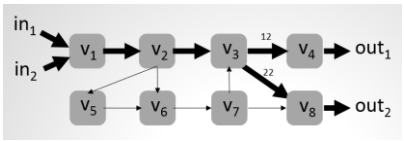
Pushdown Automaton
and **Prefix**
Rewriting Systems

Our Approach:

Automated Whatif Analysis



FT	In-I	In-Label	Out-I	op
τ_{v_1}	in_1	$\perp$	(v_1, v_2)	$push(10)$
	in_2	$\perp$	(v_1, v_2)	$push(20)$
τ_{v_2}	(v_1, v_2)	10	(v_2, v_3)	$swap(11)$
	(v_1, v_2)	20	(v_2, v_3)	$swap(21)$
τ_{v_3}	(v_2, v_3)	11	(v_3, v_4)	$swap(12)$
	(v_2, v_3)	21	(v_3, v_8)	$swap(22)$
	(v_7, v_3)	11	(v_3, v_4)	$swap(12)$
	(v_7, v_3)	21	(v_3, v_8)	$swap(22)$
τ_{v_4}	(v_3, v_4)	12	out_1	pop
τ_{v_5}	(v_2, v_5)	40	(v_5, v_6)	pop
τ_{v_6}	(v_2, v_6)	30	(v_6, v_7)	$swap(31)$
	(v_5, v_6)	30	(v_6, v_7)	$swap(31)$
	(v_5, v_6)	61	(v_6, v_7)	$swap(62)$
	(v_5, v_6)	71	(v_6, v_7)	$swap(72)$
τ_{v_7}	(v_6, v_7)	31	(v_7, v_3)	pop
	(v_6, v_7)	62	(v_7, v_3)	$swap(11)$
	(v_6, v_7)	72	(v_7, v_8)	$swap(22)$
τ_{v_8}	(v_3, v_8)	22	out_2	pop
	(v_7, v_8)	22	out_2	pop



local FFT	Out-I	In-Label	Out-I	op
τ_{v_2}	(v_2, v_3)	11	(v_2, v_6)	$push(30)$
	(v_2, v_3)	21	(v_2, v_6)	$push(30)$
	(v_2, v_6)	30	(v_2, v_5)	$push(40)$
global FFT	**Out-I**	**In-Label**	**Out-I**	**op**
τ'_{v_2}	(v_2, v_3)	11	(v_2, v_6)	$swap(61)$
	(v_2, v_3)	21	(v_2, v_6)	$swap(71)$
	(v_2, v_6)	61	(v_2, v_5)	$push(40)$
	(v_2, v_6)	71	(v_2, v_5)	$push(40)$

Router **configurations**
(Cisco, Juniper, etc.)

Compilation

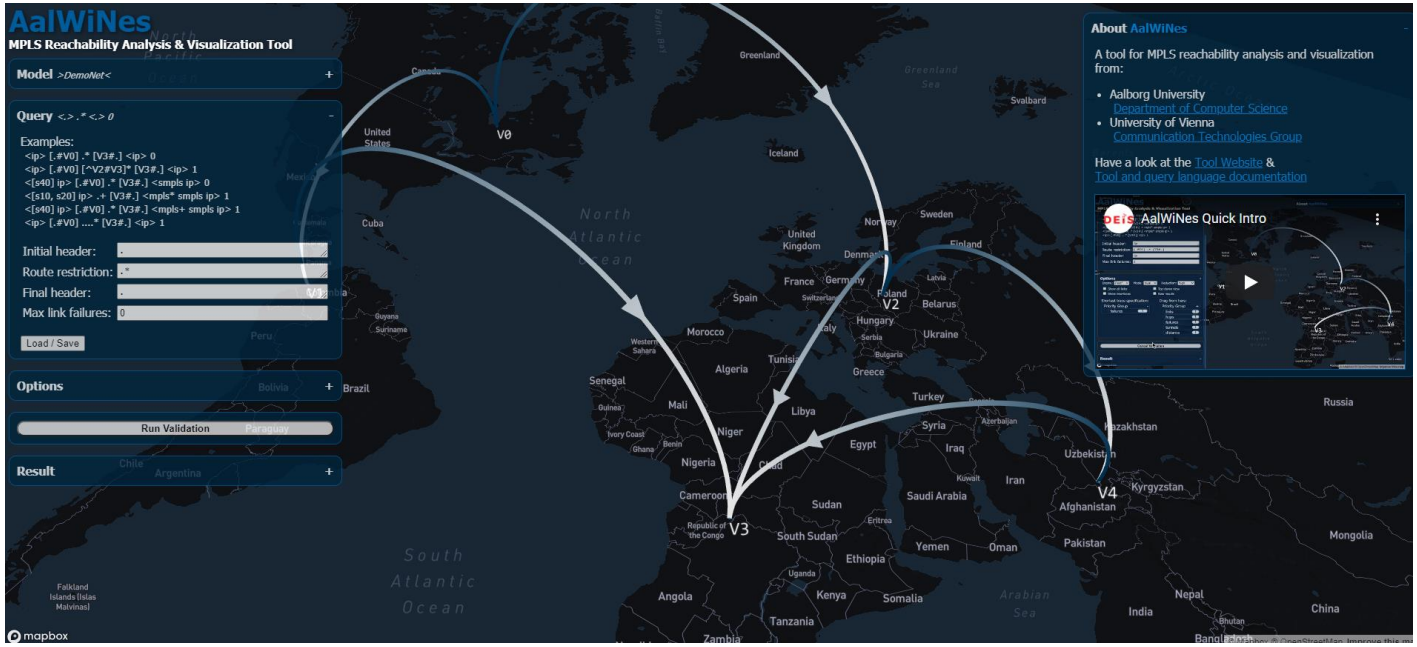
Interpretation

$pX \Rightarrow qXX$
$pX \Rightarrow qYX$
$qY \Rightarrow rYY$
$rY \Rightarrow r$
$rX \Rightarrow pX$

Pushdown Automaton
and **Prefix**
Rewriting Systems

Perdiodic or on-demand check of policy compliance, in polynomial time (Theorem by Büchi). First insights into self-repairing configurations!

P-Rex / AalWiNes Tool

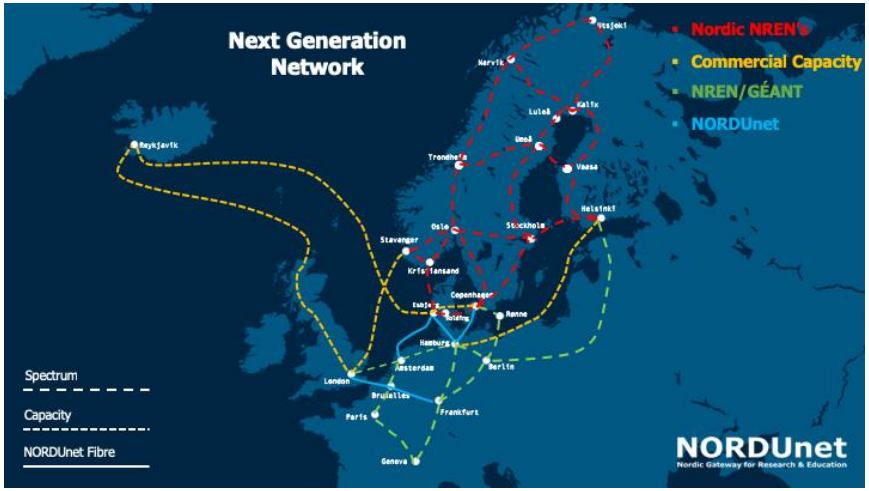


Tool: https://demo.aalwines.cs.aau.dk/

Youtube: https://www.youtube.com/watch?v=mvXAn9i7_Q0

Case Study: NORDUnet

- Regional service provider
- 24 MPLS routers geographically distributed
- For most queries of operators: answer within seconds



Project Examples

MARS: Near-Optimal Throughput with Shallow Buffers in Reconfigurable Datacenter Networks*

VAMSI ADDANKI, Faculty of Electrical Engineering and Computer Science, TU Berlin, Germany
CHEN AVIN, School of Electrical and Computer Engineering, Ben-Gurion University of the Negev, Israel
STEFAN SCHMID, Faculty of Electrical Engineering and Computer Science, TU Berlin, Germany

The performance of large-scale computing systems often critically depends on high-performance communication networks. Dynamically reconfigurable topologies, e.g., based on optical circuit switches, are emerging as an innovative new technology to deal with the explosive growth of datacenter traffic. Specifically, *periodic* reconfigurable datacenter networks (RDCNs) such as RotorNet (SIGCOMM 2017), Opera (NSDI 2020) and Sirius (SIGCOMM 2020) have been shown to provide high throughput, by emulating a *complete graph* through fast periodic circuit switch scheduling.

However, to achieve such a high throughput, existing reconfigurable network designs pay a high price: in terms of potentially high delays, but also, as we show as a first contribution in this paper, in terms of the high buffer requirements. In particular, we show that under buffer constraints, emulating the high-throughput complete graph is infeasible at scale, and we uncover a spectrum of unvisited and attractive alternative RDCNs, which emulate regular graphs, but with lower node degree than the complete graph.

We present MARS, a periodic reconfigurable topology which emulates a d-regular graph with near-optimal throughput. In particular, we systematically analyze how the degree d can be optimized for throughput given the available buffer and delay tolerance of the datacenter. We further show empirically that MARS achieves higher throughput compared to existing systems when buffer sizes are bounded.

Challenging the Need for Packet Spraying in Large-Scale Distributed Training

Vamsi Addanki
TU Berlin
Berlin, Germany

Prateesh Goyal
Microsoft Research
Redmond, USA

Ilias Marinos
Microsoft Research
Redmond, USA

Abstract

Large-scale distributed training in production datacenters constitutes a challenging workload bottlenecked by network communication. In response, both major industry players (e.g., Ultra Ethernet Consortium) and parts of academia have surprisingly, and almost unanimously, agreed that packet spraying is *necessary* to improve the performance of large-scale distributed training workloads.

In this paper, we challenge this prevailing belief and pose the question: *How close can a singlepath transport approach an optimal multipath transport?* We demonstrate that singlepath transport (from a NIC's perspective) is sufficient and can perform nearly as well as an ideal multipath transport with packet spraying, particularly in the context of distributed training in leaf-spine topologies. Our assertion is based on four key observations about workloads driven by collective communication patterns: *(i)* flows within a collective start almost simultaneously, *(ii)* flow sizes are nearly equal, *(iii)* the completion time of a collective is more crucial than individual flow completion times, and *(iv)* flows can be split upon arrival. We analytically prove that singlepath transport, using minimal flow splitting (at the application layer), is equivalent to an ideal multipath transport with packet spraying in terms of maximum congestion. Our preliminary evaluations support our claims. This paper suggests an alternative agenda

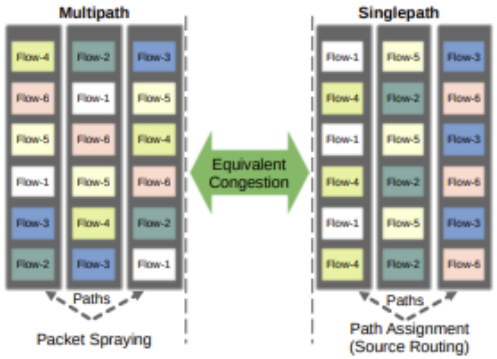


Figure 1: In contrast to traditional datacenter workloads, distributed training workloads exhibit certain properties in terms of flow sizes, the number of concurrent flows, and arrival times that allow singlepath transport to achieve nearly the same performance as an optimal multipath transport. The problem essentially boils down to assigning paths to each flow in order to minimize congestion.

In the wake of this emerging problem, both industry and academia have almost unanimously agreed that multipath transport is *necessary* to improve the performance of large-scale distributed training workloads [3, 9]. Multipath trans-

Credence: Augmenting Datacenter Switch Buffer Sharing with ML Predictions

Vamsi Addanki
TU Berlin

Maciej Pacut
TU Berlin

Stefan Schmid
TU Berlin

Abstract

Packet buffers in datacenter switches are shared across all the switch ports in order to improve the overall throughput. The trend of shrinking buffer sizes in datacenter switches makes buffer sharing extremely challenging and a critical performance issue. Literature suggests that push-out buffer sharing algorithms have significantly better performance guarantees compared to drop-tail algorithms. Unfortunately, switches are unable to benefit from these algorithms due to lack of support for push-out operations in hardware. Our key observation is that drop-tail buffers can emulate push-out buffers if the future packet arrivals are known ahead of time. This suggests that augmenting drop-tail algorithms with predictions about the future arrivals has the potential to significantly improve performance.

This paper is the first research attempt in this direction. We propose CREDENCE, a drop-tail buffer sharing algorithm augmented with machine-learned predictions. CREDENCE can unlock the performance only attainable by push-out algorithms so far. Its performance hinges on the accuracy of predictions. Specifically, CREDENCE achieves near-optimal performance of the best known push-out algorithm LQD (Longest Queue Drop) with perfect predictions, but *gracefully* degrades to the

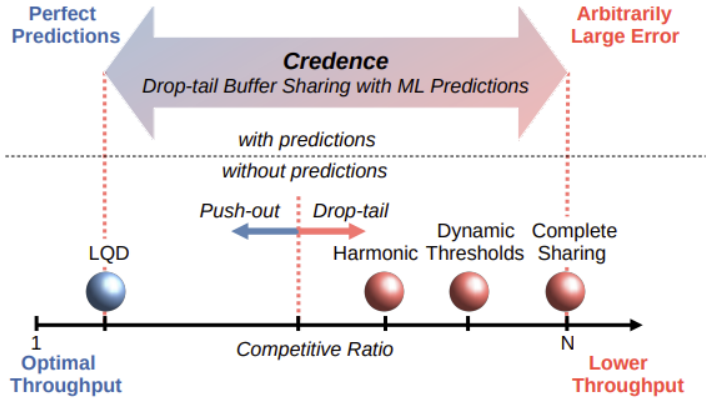


Figure 1: Augmenting drop-tail buffer sharing with ML predictions has the potential to significantly improve throughput compared to the best possible drop-tail algorithm (without predictions), and unlock the performance that was only attainable by push-out so far.

proportional to capacity increase [13]. As a result, the buffer available per port per unit capacity of datacenter switches has been gradually reducing over time. Worse yet, datacenter traffic is bursty even at microsecond timescales [22, 57]. This

PowerTCP: Pushing the Performance Limits of Datacenter Networks*

Vamsi Addanki
University of Vienna
TU Berlin

Oliver Michel
University of Vienna
Princeton University

Stefan Schmid
University of Vienna
TU Berlin

Abstract

Increasingly stringent throughput and latency requirements in datacenter networks demand fast and accurate congestion control. We observe that the reaction time and accuracy of existing datacenter congestion control schemes are inherently limited. They either rely only on explicit feedback about the network state (e.g., queue lengths in DCTCP) or only on variations of state (e.g., RTT gradient in TIMELY). To overcome these limitations, we propose a novel congestion control algorithm, PowerTCP, which achieves much more fine-grained congestion control by adapting to the bandwidth-window product (henceforth called power). PowerTCP leverages in-band network telemetry to react to changes in the network instantaneously without loss of throughput and while keeping queues short. Due to its fast reaction time, our algorithm is particularly well-suited for dynamic network environments and bursty traffic patterns. We show analytically and empirically that PowerTCP can significantly outperform the state-of-the-art in both traditional datacenter topologies and emerging reconfigurable datacenters where frequent bandwidth changes make congestion control challenging. In traditional datacenter networks, PowerTCP reduces tail flow completion times of short flows by 80% compared to DCQCN and TIMELY, and by 33% compared to HPCC even at 60% network load. In reconfigurable datacenters, PowerTCP achieves 85% circuit utilization without incurring additional latency and cuts tail latency by at least 2x compared to existing approaches.

stringent performance requirements are introduced by today's trend of resource disaggregation in datacenters where fast access to remote resources (e.g., GPUs or memory) is pivotal for the overall system performance [36]. Building systems with strict performance requirements is especially challenging under bursty traffic patterns as they are commonly observed in datacenter networks [12, 16, 47, 53, 55].

These requirements introduce the need for fast and accurate network resource management algorithms that optimally utilize the available bandwidth while minimizing packet latencies and flow completion times. Congestion control (CC) plays an important role in this context being ''a key enabler (or limiter) of system performance in the datacenter'' [34]. In fact, fast reacting congestion control is not only essential to efficiently adapt to bursty traffic [29,48], but is also becoming increasingly important in the context of emerging reconfigurable datacenter networks (RDCNs) [13,14,20,33,38,39,50]. In these networks, a congestion control algorithm must be able to quickly ramp up its sending rate when high-bandwidth circuits become available [43].

Traditional congestion control in datacenters revolves around a bottleneck link model: the control action is related to the state i.e., queue length at the bottleneck link. A common goal is to efficiently control queue buildup while achieving high throughput. Existing algorithms can be broadly classified into two types based on the feedback that they react to. In the following, we will use an analogy to electrical circuits[1] to describe these two types. The first category of

Reverie: Low Pass Filter-Based Switch Buffer Sharing for Datacenters with RDMA and TCP Traffic

Vamsi Addanki
TU Berlin

Wei Bai
Microsoft Research

Stefan Schmid
TU Berlin

Maria Apostolaki
Princeton University

Abstract

The switch buffers in datacenters today are dynamically shared by traffic classes with different loss tolerance and reaction to congestion signals. In particular, while legacy applications use loss-tolerant transport, e.g., DCTCP, newer applications require lossless datacenter transport, e.g., RDMA over Converged Ethernet. Unfortunately, as we analytically show in this paper, the buffer-sharing practices of today's datacenters pose a fundamental limitation to effectively *isolate* RDMA and TCP while also maximizing *burst absorption*. We identify two root causes: *(i)* the buffer-sharing for RDMA and TCP relies on two independent and often conflicting views of the buffer, namely ingress and egress; and *(ii)* the buffer-sharing scheme micromanages the buffer and overreacts to the changes in its occupancy during transient congestion.

At a high level, the goal of a buffer-sharing scheme is to provide isolation between traffic classes, while maximizing the benefit of the buffer e.g., by absorbing bursts and achieving high throughput. Existing buffer management schemes (even recent ones) [1, 8, 15, 25] were designed considering exclusively loss-tolerant traffic (e.g., TCP variants). However, modern datacenters host traffic classes with different loss tolerance. Concretely, along with traditional loss-tolerant transport protocols, many clouds, e.g., Azure [11], Alibaba [22] and OCI [38], deploy RDMA over Converged Ethernet which requires lossless transport. In order to guarantee zero packet loss for RDMA, production datacenters enable Priority Flow Control (PFC) at the switches [11].

The co-existence of TCP and RDMA traffic in the switch buffer makes sharing the buffer particularly challenging. While, in principle, TCP and RDMA traffic have the same perfor-

ABM: Active Buffer Management in Datacenters

Vamsi Addanki*
TU Berlin

Maria Apostolaki*
Princeton University

Manya Ghobadi
MIT

Stefan Schmid
TU Berlin

Laurent Vanbever
ETH Zurich

ABSTRACT

Today's network devices share buffer across queues to avoid drops during transient congestion and absorb bursts. As the buffer-per-bandwidth-unit in datacenter decreases, the need for optimal buffer utilization becomes more pressing. Typical devices use a hierarchical packet admission control scheme: First, a Buffer Management (BM) scheme decides the maximum length per queue at the device level and then an Active Queue Management (AQM) scheme decides which packets will be admitted at the queue level. Unfortunately, the lack of cooperation between the two control schemes leads to *(i)* harmful interference across queues, due to the lack of isolation; *(ii)* increased queueing delay, due to the obliviousness to the per-queue drain time; and *(iii)* thus unpredictable burst tolerance. To overcome these limitations, we propose ABM, Active Buffer Management which incorporates insights from both BM and AQM. Concretely, ABM accounts for both total buffer occupancy (typically used by BM) and queue drain time (typically used by AQM). We analytically prove that ABM provides isolation, bounded buffer drain time and achieves predictable burst tolerance without sacrificing throughput. We empirically find that ABM improves the 99th percentile FCT for short flows by up to 94% compared to the state-of-the-art buffer management. We further show that ABM improves the performance of advanced datacenter transport proto-

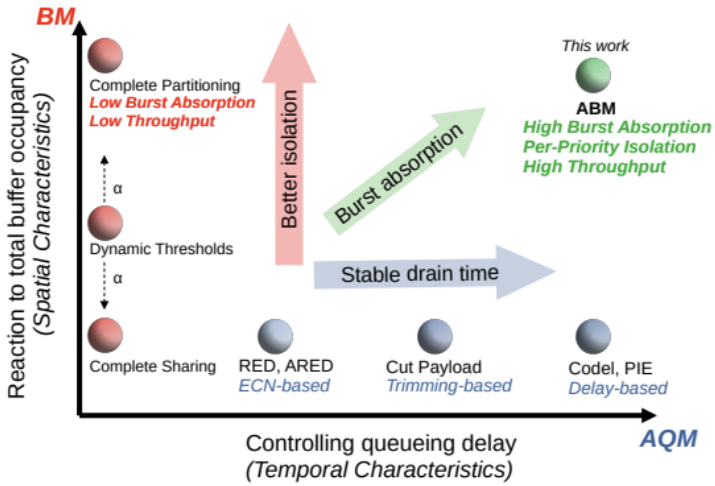


Figure 1: BM and AQM are orthogonal in their goals, and the hierarchical scheme fundamentally limits the burst absorption capabilities of the buffer.

1 INTRODUCTION

Network devices are equipped with a buffer to avoid drops during

TCP's Third Eye: Leveraging eBPF for Telemetry-Powered Congestion Control

Jörn-Thorben Hinz
TU Berlin

Vamsi Addanki
TU Berlin

Csaba Györgyi
University of Vienna

Theo Jepsen
Intel

Stefan Schmid
TU Berlin

ABSTRACT

For years, congestion control algorithms have been navigating in the dark, blind to the actual state of the network. They were limited to the course-grained signals that are visible from the OS kernel, which are measured locally (e.g., RTT) or hints of imminent congestion (e.g., packet loss and ECN). As applications and OSs are becoming ever more distributed, it is only natural that the kernel have visibility beyond the host, into the network fabric. Network switches already collect telemetry, but it has been impractical to export it for the end-host to react.

Although some telemetry-based solutions have been proposed, they require changes to the end-host, like custom hardware or new protocols and network stacks. We address the challenges of efficiency and protocol compatibility, showing that it is possible *and practical* to run telemetry-based congestion control algorithms in the kernel. We designed a framework that uses eBPF to run CCAs that can execute different control laws by selecting different types of telemetry. It can be deployed in brownfield environments, without requiring all switches be telemetry-enabled, or kernel recompilation at the end-hosts. When our eBPF program is deployed on hosts without hardware or OS changes, TCP incast workloads experience less queuing (thus lower latency), faster convergence and better fairness.

1 INTRODUCTION

The volume of traffic in datacenters is increasing rapidly over time [6, 31, 33]. The throughput and latency offered by the underlying architecture and the set of protocols plays a critical role in the performance of modern cloud-based applications [26]. To this end, major research efforts over the past decade have been in two main domains: hardware offloading [2, 3, 20] and advanced congestion control [1, 18, 25, 26, 28, 42].

On the one hand, offloading computationally heavy tasks to hardware reduces software overheads but it comes at the cost of programmability and flexibility of the network stack, requiring specialized hardware such as RDMA (Remote Direct Memory Access) NICs. On the other hand, advanced congestion control offers immense benefits in-terms of throughput and latency, but it comes at the cost of bandwidth and computational overheads [12, 27]. These tradeoffs are clearly visible in today's datacenters, which rely on traditional TCP/IP for storage applications [19, 21]. Even in large-scale datacenters with RDMA capabilities, traditional TCP/IP traffic still accounts for up to 30% of the total traffic [5].

In this paper, we lay the groundwork for flexible and low-overhead telemetry-based congestion control algorithms (CCAs) in datacenter networks. Recent advancements in networking have finally made this possible *and practical*. The network data-plane is now programmable, both at the end-hosts and in the network fabric.

Baiji: Domain Planning for CDNs under the 95th Percentile Billing Model

Juan Vanerio*†, Huiran Liu‡, Qi Zhang‡ and Stefan Schmid‡§*
*Faculty of Computer Science, University of Vienna, Austria
†AIT Austrian Institute of Technology, Austria
‡INET, Faculty IV (EECS), Technical University Berlin, Germany
§Fraunhofer SIT, Germany

***Abstract*—Content Distribution Networks (CDNs) play a crucial role in efficiently delivering online content to end-users. In this paper, we initiate the study of CDN domain planning with flexible assignments of domains to Points of Presence (PoPs) within a CDN, with the objective of minimizing the cost of transmissions while providing sufficient resources to serve the communication demands. The problem is subject to practical constraints of network deployment such as a percentile-based billing model, PoP's bandwidth and committed rate limits, geographic locality and quota constraints and minimum per domain cache-hit-ratios.**

We formulate the problem as an offline optimization task with a nonlinear objective function and linear constraints, which becomes computationally intensive for medium-sized instances. The 95th percentile billing model, commonly used by service providers, contributes significantly to this non-linearity. To address this, we propose *Baiji*, a multi-algorithm approach leveraging insights from our formulation

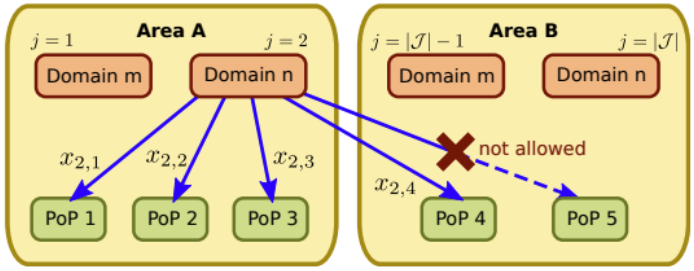


Fig. 1. **An instance of the CDN domain planning problem.** Domain names n and m must be served to users on areas A and B, containing resp. three and two PoPs. The areas and domains form domain-area pairs (indexed by j). PoP 5 is forbidden to serve domain n in area A. Attributes x_{ji} indicate the demand fraction for pair j served from PoP i.

videos. These interconnected PoPs form an intricate network

Starlink Performance through the Edge Router Lens

Sarah-Michelle Hammer
TU Berlin
Berlin, Germany

Vamsi Addanki
TU Berlin
Berlin, Germany

Max Franke
TU Berlin
Berlin, Germany

Stefan Schmid
TU Berlin
Berlin, Germany

ABSTRACT

Low-Earth Orbit satellite-based Internet has become commercially available to end users, with Starlink being the most prominent provider. Starlink has been shown to exhibit a periodic pattern with a characteristic throughput drop on the boundaries of 15s intervals. A multitude of prior works hypothesize various root causes for this pattern, such as reordering and packet loss. Some works have attributed these effects to the edge router, advocating for explicit feedback to the transport layer. However, with the edge router being a proprietary Starlink device, it raises questions about the extent of its influence on periodic throughput drops, losses, and jitter, leaving us to wonder if we fully understand the underlying issues.

This paper presents the first measurement study with a vantage point that is by far the closest (last hop) to the core Starlink network. We use a Generation 1 dish, which allows us to bypass the proprietary Starlink router and connect a Linux server directly to the dish. We investigate the impact of the edge router on the observed periodic pattern in Starlink performance. Our results are primarily negative in terms of any significant buffer buildup and packet losses at the edge router, suggesting that the causality of the observed patterns lies entirely in the core network, a proprietary space that cannot be fixed by the end user. Interestingly, we observe similar patterns even with a constant bitrate UDP sender, likely indicating that the periodic drop in throughput is not an inherent limitation of existing TCP implementations but rather a core network characteristic!

CCS CONCEPTS

• **Networks** → **Network measurement**; *Network performance analysis.*

KEYWORDS

Starlink, Low-Earth Orbit Satellite, Internet measurements

ACM Reference Format:
Sarah-Michelle Hammer, Vamsi Addanki, Max Franke, and Stefan Schmid. 2024. Starlink Performance through the Edge Router Lens. In *2nd International Workshop on LEO Networking and Communication (LEO-NET 24), November 18–22, 2024, Washington D.C., DC, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3697253.3697273

1 INTRODUCTION

The first wide-area packet-switched network dates back to the late 1960s [9], and it took more than five decades for the internet to grow from a few computers at universities on the West Coast to a billion devices connected across the globe. Several breakthroughs in technology have enabled this growth, including the development of the TCP/IP protocol suite, broadband connectivity to end-user homes, and wireless technologies such as WiFi and cellular networks [9]. Yet, even after five decades of technological advancements, providing high-speed internet access to all corners of the world remains an economic and technical challenge[3].

Low-Earth Orbit (LEO) satellite networks have emerged as a promising solution to this challenge, offering internet access to remote areas, even during disasters [19]. Service

Conclusion

⇢ Our main focus: self-adjusting networks

⇢ Example 1: demand-aware topology

⇢ Example 2: policy-compliant networks

- ⇀ self-verifying
- ⇀ self-repairing

⇢ On both fronts: tip of the iceberg!

Thank you!